

AMSAA TM 98



AMSAA

AD

TECHNICAL MEMORANDUM NO. 98

THE FIRST CONFERENCE OF USERS OF
THE MAGIC AND SAM-C PROGRAMS

R. A. Marking

May 1971

Approved for public release,
distribution unlimited.

U.S. ARMY ARTILLERY RESEARCH AND DEVELOPMENT CENTER
ARMY MATERIAL SYSTEMS ANALYSIS AGENCY
APPROVED PROVING GROUND, MARYLAND

DISPOSITION:

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position.

WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the AEC position unless so specified.

WHITE SOUTHERN
POST OFFICE
RECEIVED
JUL 10 1964
AIR MAIL

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) US Army Aberdeen Research and Development Center Army Materiel Systems Analysis Agency Aberdeen Proving Ground, Maryland 21005		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP	
3. REPORT TITLE THE FIRST CONFERENCE OF USERS OF THE MAGIC AND SAM-C PROGRAMS			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates)			
5. AUTHOR(S) (First name, middle initial, last name) R. A. Marking			
6. REPORT DATE May 1971		7a. TOTAL NO. OF PAGES 153	7b. NO. OF REFS
8a. CONTRACT OR GRANT NO.		8a. ORIGINATOR'S REPORT NUMBER(S)	
b. PROJECT NO. RDT&E 1T562601A259		AMSAA TM NO. 98	
c.		8b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.			
10. DISTRIBUTION STATEMENT Approved for public release, distribution unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY US Army Materiel Command Washington, D.C. 20315	
13. ABSTRACT <p>This report contains the proceedings of the first conference of the users of the MAGIC and SAM-C computer programs held August 6, 7, and 8, 1969, at the Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Maryland. The purpose of the conference was to determine which subprograms constitute the MAGIC and SAM-C programs, to create program source decks, to create benchmark test problems, and to discuss problems of mutual interest.</p> <p>Also included are the addresses, in expanded form, of the principal users of the two programs. These addresses included corrections and improvements to the two codes.</p>			

DD FORM 1473

NOV 65

REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Nuclear fluence, flux, dose, or dose rate analysis Radiation transport benchmark problems Target description benchmark problems MAGIC Conference SAM-C Conference						

UNCLASSIFIED

Security Classification

ARMY MATERIEL SYSTEMS ANALYSIS AGENCY

TECHNICAL MEMORANDUM NO. 98

THE FIRST CONFERENCE OF USERS OF
THE MAGIC AND SAM-C PROGRAMS

R. A. Marking

MAY 1971

Approved for public release,
distribution unlimited.



RDT&E Project No. 1T562601A259

ABERDEEN PROVING GROUND, MARYLAND

ARMY MATERIEL SYSTEMS ANALYSIS AGENCY

TECHNICAL MEMORANDUM NO. 98

RAMarking/paw
Aberdeen Proving Ground, Md.

THE FIRST CONFERENCE OF USERS OF
THE MAGIC AND SAM-C PROGRAMS

ABSTRACT

This report contains the proceedings of the first conference of the users of the MAGIC and SAM-C computer programs held August 6, 7, and 8, 1969, at the Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Maryland. The purpose of the conference was to determine which subprograms constitute the MAGIC and SAM-C programs, to create program source decks, to create benchmark test problems, and to discuss problems of mutual interest.

Also included are the addresses, in expanded form, of the principal users of the two programs. These addresses included corrections and improvements to the two codes.

CONTENTS

	Page
ABSTRACT	3
1. INTRODUCTION AND PURPOSE	7
1.1 Copy of Conference Announcement (Abridged)	7
1.2 Opening Address	11
2. INTRODUCTION OF PARTICIPANTS	12
3. RECENT ACTIVITY WITH MAGIC AND SAM-C	22
3.1 Recent Activity with MAGIC at AMSAA	22
3.1.1 Changes to MAGIC	22
3.1.2 Proposed Plans for MAGIC	27
3.1.3 Concluding Remarks	28
3.1.4 Additional Information	28
3.2 Recent Activity with SAM-C at NEL	34
3.2.1 Recent Calculations	34
3.2.2 Corrections/Improvements	35
3.2.3 Improvements in Progress	36
3.2.4 Planned Improvements	36
3.3 Coordination of MAGIC and SAM-C	36
3.3.1 Discussion	36
3.3.2 Program Changes	38
4. DEFINITION OF MAGIC AND SAM-C; CREATION OF SOURCE DECKS	38
4.1 MAGIC	38
4.1.1 Standard Version	38
4.1.2 Ad Hoc Problems	40
4.2 SAM-C	40
4.2.1 Standard Version	40
4.2.2 Ad Hoc Problems	40
5. CREATION OF BENCHMARK PROBLEMS	41
5.1 MAGIC	41
5.2 SAM-C	41
APPENDIX A MAGIC/SAM-C CONFERENCE PARTICIPANTS	43
APPENDIX B MAGI REPORT MR 6902 (ABRIDGED)	45
APPENDIX C SAM-C BENCHMARK PROBLEM INPUT	59
APPENDIX D LISTING OF THE AMSAA OCTOBER REVISED STANDARD MAGIC PROGRAM	73
DISTRIBUTION LIST	151

Next page is blank.

THE FIRST CONFERENCE OF USERS OF
THE MAGIC AND SAM-C PROGRAMS

1. INTRODUCTION AND PURPOSE

This report presents the proceedings of the first large-scale conference of the users of the MAGIC and SAM-C computer programs. The conference convened at 0915 on August 6, 1969, in Building 328 at Aberdeen Proving Ground, Maryland. Early in the afternoon of the 7th the conferees divided themselves into two groups: those interested in the MAGIC code and those interested in the SAM-C. The conference closed shortly after noon on Friday the 8th.

The following sections contain abridged versions of the announcement letter and of the opening address.

1.1 Copy of Conference Announcement (Abridged).

"AMXRD-AWF

"SUBJECT: Conference for Users of the MAGIC/SAM-C Computer Programs

"1. References:

a. The MAGIC SAM-C Target Analysis Technique, AMSAA Technical Reports 4, 10, 11, 13, and 14.

b. A Geometric Description Technique Suitable for Computer Analysis of Both the Nuclear and Conventional Vulnerability of Armored Military Vehicles, MAGI-6701 (AD 847576).

c. UNC-SAM-2: A FORTRAN Monte Carlo Program Treating Time-Dependent Neutron and Photon Transport Through Matter, UNC-5157 (AD 647470).

"2. The SAM-C Monte Carlo radiation transport program and the MAGIC conventional projectile ray-tracing program were developed by MAGI (Mathematical Applications Group, Inc) under Contract No. DAAD05-67-C-0041 for this agency (ref. a). MAGIC is the computer code that assembles data along selected rays through a target by employing the MAGI originated Combinatorial Geometry technique (ref. b) which utilizes combinations of certain basic solids such as boxes, wedges, etc., to

describe the target. SAM-C is the UNC-SAM-2 Monte Carlo nuclear radiation transport code (ref. c) with the original geometry routines replaced by those of the Combinatorial Geometry technique. These programs have been disseminated to a number of interested government agencies for their use and will be made available to others upon request.

"5. Recent activity with MAGIC has resulted in several advances including the addition of new "library" solids, additional input checking, and faster input processing. These advances are clearly applicable to SAM-C as well and are thus of general interest to both groups. On the other hand, recent activity with SAM-C indicates that the flux-at-a-point routine is incorrectly coded, versions for different computing systems have non-trivial differences in logic/organization, and the computational procedure for carrying out a complete set of calculations is unacceptably complicated.

"4. As a result of the many questions, comments, and suggestions from the personnel involved in using SAM-C/MAGIC, it is clear that everyone involved can benefit from an informal discussion of problem areas and a concerted attack on areas of mutual interest. It is the purpose of this letter to announce an informal, unclassified conference to define and solve problems in the MAGIC and SAM-C computer codes developed under the auspices of this agency.

"5. It is envisioned that the conference will begin on 6 August 1969 at 0900 hours in the Conference Room of ARDC Building 328 under the chairmanship of Mr. R. A. Marking of AMSAA. At this writing it is envisioned that 2 days will be required with a third day allotted only to provide a buffer period.

"6. The first day will be devoted to introductory, informal discussions of individual problems, solutions, and changes to the various codes plus presentations (on MAGIC by Mr. Larry Bain, Methodology Office, AMSAA, and on SAM-C by Dr. Wayne Coleman, Nuclear Physics Branch, NEL) of information gathered too late for inclusion into the reports of reference a. The item, "Introduction of Participants," is envisioned as an opportunity for each participant to identify which program(s) his agency is using, what problems have been encountered, any solutions that have been created, and what specific problem areas he would like to see addressed during the conference period. About 20 minutes will be allotted for each individual.

"7. The second day is planned for the creation of ad hoc working groups to achieve solutions of the problems defined on the first day. An additional day is scheduled to allow an orderly conclusion of the working group projects and the conference as a whole if necessary; the form and content of a conference report will be decided on the last day.

"8. Two items have been of special interest to all those contacted thus far: (1) the establishment of FORTRAN source decks for MAGIC and SAM-C that are not subject to the vagaries of day-to-day changes and are available to be copied by new user agencies or in the event that serious problems develop with an existing source deck; and (2) the creation of "benchmark" test problems that will provide valid tests of all features/options operational using the "archival" or "library" source deck. It is expected that these two items can be introduced in the afternoon of the first day of the conference.

"9. Participation in the conference is encouraged to promote the utility of the SAM-C and MAGIC programs. Representation from your organization is invited. The distribution list is not considered exhaustive and interested personnel within government may be invited to attend by contacting Mr. Marking. Participants are encouraged to bring listings of the current working versions of their program(s).

"10. It is intended that the meeting will result in documented improvements in the SAM-C and MAGIC source programs and/or implementation procedures. This documentation is expected to be in the form of a letter or technical note and copies will be sent to all participants.

"11. The desirability of scheduling similar meetings at regular intervals will be discussed as a means of maintaining lines of communications between MAGIC-SAM-C users."

12. (Administrative)

13. (Administrative)

FOR THE DIRECTOR:

/s/ Morgan G. Smith
MORGAN G. SMITH
Chief, Ground Warfare Division

1 Incl
as
(CF)

PROPOSED CONFERENCE AGENDA

DAY 1

"Introduction and Purpose

Introduction of Participants

Recent AMSAA Activity with MAGIC

Recent NEL Activity with SAM-C

Definition of MAGIC and SAM-C Capabilities and

Creation of Appropriate Source Decks

Creation of Benchmark Problems

DAY 2

Conclusion of the Creation of Archival Source
Decks and Benchmark Problems

Formation of Sub-groups to Solve the Problems
Defined on Day 1

Discussion and Drafting of Documentation

Consideration of Future Meetings

DAY 3 (If Necessary)

Conclusion of Unfinished Business"

Incl 1 to 1tr

1.2 Opening Address.

"Welcome to the 1st Conference of Users of the MAGIC and SAM-C Programs.

"As stated in the Conference letter-announcement, this is to be an informal, unclassified conference of the users of the MAGIC and/or SAM-C computer programs (as well as any of the peripheral programs that might be of mutual interest).

"The purpose of this conference is three-fold:

- to find out where we stand with respect to the actual operation of both SAM-C and MAGIC
 - who is using what
 - on what machines are the codes operating
 - what changes of substance (e.g., packing into 36-bit v. 48-bit words) have been made
 - what kinds of problem areas are being considered (e.g., deep penetration as opposed to close-in transport problems, penetrator fragmentation, x-ray or thermal neutron problems, etc.)
 - what sort of functional and/or theoretical problems are being encountered in operating these codes (e.g., functional identifiers of "O", incorrect evaluation of the uncollided flux, improper coding of SAM-C for inelastic scattering, etc.)
- to define what SAM-C and MAGIC should be capable of
 - there are a number of versions of SAM-C several of which are different enough to require separate operating manuals
 - changes have significantly increased the speed of MAGIC thus making here-to-fore impractical applications worth considering
 - should SAM-C be modified along the lines of UNC-SAM-3 (ENDF/B cross-sections and a non-common energy mesh)
 - should MAGIC calculate vulnerable areas
 - should the geometry processing routines (e.g., GENI, RPPIN, ALBERT, etc.) be called MAGIC and the "driving" or controlling routines such as VOLUM and GRID be handled as separate "packages"
 - should MAGIC employ packing and what effect would its absence have on SAM-C

- to create benchmark problems
 - compatibility of geometric input between MAGIC and SAM-C
 - should they be mathematical tests of all the options or tests of physical acceptability of some combination of both...."

2. INTRODUCTION OF PARTICIPANTS

At the beginning of the Conference the individual participants indicated (1) which program they were using, (2) on which computers was it routinely used (or intended to be used), (3) what were the nature and complexity of the problems treated, (4) what program innovations had been made, and (5) what problems or errors had been encountered. Additionally, it was requested that participants indicate specific program problems to discuss during the Conference.

To accomplish these introductions in an orderly manner, a form covering all of the obvious points of interest was used. Since some of the agencies were represented by more than one person, representatives from the same group using the same program caucused to present a unified picture of their work and problems.

The participants' introductions follow the list of agencies; they are in alphabetical order.

List of Agencies Represented

Aeronautical Systems Division
(Wright-Patterson)

Air Force Armament Laboratory
(Eglin)

Air Force Weapons Laboratory
(Kirtland)

Ground Warfare Division (AMSAA)
(Aberdeen)

Methodology Office (AMSAA)
(Aberdeen)

Naval Weapons Laboratory*
(Dahlgren)

Nuclear Effects Laboratory (BRL)*
(Edgewood)

* No Participant Introduction Form available.

List of Agencies Represented (Cont'd)

SMUPA-DW6*
(Picatinny)

SMUPA-SS
(Picatinny)

SMUPA-TW3
(Picatinny)

SMUPA-VC1
(Picatinny)

RSIC (ORNL)
(Oak Ridge)

Signature & Propagation Laboratory**
(Aberdeen)

Terminal Ballistics Laboratory (BRL)*
(Aberdeen)

Vulnerability Laboratory (BRL)
(Aberdeen)

* No Participant Introduction Form available.

** Observer only, no Participant Introduction Form included.

PARTICIPANT INTRODUCTION FORM

Agency: USAF, ASD (ASBRS), WPAFB, Ohio 45433

Name(s) of Representative(s): Gerald Bennett (ASBRS)
Roy Hilbrand (ASVCP)

Program Used: MAGIC

Purpose: To provide target descriptions for use in aircraft vulnerability analyses.

Computer(s) Used: Name IBM Direct Coupled 7044/7094 Word Size 36 bits

Memory Size: Total 32768 Available Unk

Tape Drives: No. 16 No. of Channels 4 1401-Compatible? Yes

Program Requirements: Storage 28K Packed Word Size 35 bits

Tape Drives 2 Links? Yes, 1

Dependence on Assembly Language None

Planned Program Usage: To generate target description for aircraft vulnerable area computation.

Planned Program Changes: Addition of plotting, presented area, and volume subroutines; modifications as required to generate and store data for efficient processing in vulnerable area computation program; further simplifications, as possible, to input descriptive data.

Program Innovations: The use of any body as a target volume subdivision (i.e., as an RPP); the streamlining of MAGIC by stripping out about 16 of the subroutines and recoding of others; restructuring of the Master-Aster array (M-A) deleting some items from the M-A array; recoding and repacking for 36 bit words, viz., 35 bits and one sign bit; changed grid cell generation; changed printout; allowing the attack plane to be outside of the enclosing volume; and disposal of random number generator requirement.

Program Problems/Errors: Core storage (too large); complexity in preparation of data. (Various program errors have been corrected and the corrected listings have been forwarded to AMSAA.)

Program Changes of Immediate Interest: Addition of an airfoil shape to the solid library; a more extensive ARB of perhaps 10 to 12 sides; introduction of "canned" standard aircraft component descriptions (e.g., a pilot).

PARTICIPANT INTRODUCTION FORM

Agency: Air Force Armament Laboratory

Name(s) of Representative(s): Sue Gibson

Program Used: MAGIC

Purpose: To be used with a vulnerable area program to produce vulnerable area program to produce vulnerable areas of foreign targets.

Computer(s) Used: Name CDC 6600 Word Size 60 bits

Memory Size: Total 100,000 Available 32,000

Tape Drives: No. 16 No. of Channels Unk 1401-Compatible? No

Program Requirements: Storage 32,000 Packed Word Size 36 bits

Tape Drives 7 Links? Yes, 4

Dependence on Assembly Language None

Planned Program Usage: Describe foreign air and ground targets in terms of line of sight data.

Planned Program Changes: Complete conversion from IBM 7094 to CDC 6600. Omit TESTG and other unnecessary subroutines to allow room for addition of new figure subroutines.

Program Innovations: Point Burst subroutine has been added and is being checked.

Program Problems/Errors: Lack of storage due to amount of storage allowed to each user, not to the total amount of storage in the CDC 6600.

Program Changes of Immediate Interest: Addition of new figures and reduction of amount of storage required.

PARTICIPANT INTRODUCTION FORM

Agency: AFWL, Kirtland AFB, New Mexico 87117

Name(s) of Representative(s): Michael J. Paul - AFWL (WLRAS)
A. Kris Widdison - AFWL (WLCP-M)

Program Used: SAM-C

Purpose: Both deep-penetration (in air) and close-in transport (concrete) problems, primarily neutrons, but including prompt and secondary gammas.

Computer(s) Used: Name CDC 6600 Word Size 60 bits

Memory Size: Total 300K (w/o extended core) Available 1Mg
(w/extended core) 325K₈ (w/o extended core)

Tape Drives: No. 10 No. of Channels 9 1401-Compatible? Unk

Program Requirements: Storage generally 120K Packed Word Size 45 or
Tape Drives 1-3 Links? None 60 bits
Dependence on Assembly Language some, but easily
changed

Planned Program Usage: Hard-rock silo configurations and state-of-art neutron and gamma transport problems.

Planned Program Changes: Complete revision of input to be more understandable and logical and easier to punch. Combined time-energy-angular dependent source input (allowing input of flux from a preliminary discrete ordinates code).

Program Innovations: Free-form reading routine (eliminates need for formatting input). Cut down flux printing by 50% by eliminating extraneous lines (e.g., ΔE). Eliminate need to change NXS = and NDQ.... cards (e.g., add a parameter, say ENDM, to end of master array (COMMON DUM (250), MASTER (30000), ENDM), then NDQ = LOCF (ENDM) - LOCF (Master)).

Program Problems/Errors: None.

Program Changes of Immediate Interest: ENDF/B cross sections, inelastic scattering improvements; graphic geometry display; better geometry checking.

PARTICIPANT INTRODUCTION FORM

Agency: AMSAA, Ground Warfare Division (GWD); Methodology Office (MO)
Aberdeen Proving Ground, Maryland 21005

Name(s) of Representative(s): L. Bain (MO)
R. Lake (GWD)
J. Brewer (GWD)

Program Used: MAGIC

Purpose: Conventional vulnerability by 4" cells and/or areas for
combat vehicles and aircraft.

Computer(s) Used: Name BRLESC I & II Word Size 68 bits

Memory Size: Total 96K Available 48K

Tape Drives: No. 8 No. of Channels 4 1401-Compatible? Yes

Program Requirements: Storage 48 Packed Word Size 30

Tape Drives 4 Links? No

Dependence on Assembly Language Yes but easily
avoided

Planned Program Usage: Conventional vulnerability of combat vehicles and
aircraft (both rotary and fixed wing types).

Planned Program Changes:

- 1) Thirty bit packing for triplets and scalars.
- 2) Shielded areas.

Program Innovations: Rewrite program flow to minimize presence of
unnecessary steps, extraneous comments, and blank cards.

Program Problems/Errors: None.

Program Changes of Immediate Interest: No genuinely pressing problems.

PARTICIPANT INTRODUCTION FORM

Agency: Picatinny Arsenal, Dover, N. J.

Name(s) or Representative(s): Robert Kesselman - VC-1
John Saarmann - VC-1
Robert Barnas - SS
John Burgio - TW3

Program Used: SAM-C NEL Version

Purpose: To obtain running version on IBM 360 for radiation transport and shielding calculations.

Computer(s) Used: Name IBM 360 Word Size 32/64 bits

Memory Size: Total Unk Available 200K

Tape Drives: No. 8 No. of Channels 9 1401-Compatible? Yes

Program Requirements: Storage 200K Packed Word Size 64 bits

Tape Drives 3 Links? Unk

Dependence on Assembly Language one subroutine

Planned Program Usage: Transport and Shielding

Planned Program Changes: In January 1970 Picatinny Arsenal will start using CDC 6500; therefore, the debugging effort on the conversion has been suspended.

Program Innovations: (See comment above.)

Program Problems/Errors: (See comment above.)

Program Changes of Immediate Interest: (See comment above.)

PARTICIPANT INTRODUCTION FORM

(Observer)

Agency: Radiation Shielding Information Center
Oak Ridge National Laboratory
P. O. Box X
Oak Ridge, Tennessee 37830

Name(s) of Representative(s): Robert W. Roussin

Program Used: SAM-C

Purpose: For distribution to anyone who wants the program. (RSIC operations are sponsored by the AEC, DASA, and NASA.)
We have the CDC 6600 version for distribution (but no 6600 at ORNL).

<u>Computer(s) Used:</u>	<u>Name</u>	<u>Word Size</u>
	<u>Memory Size:</u>	<u>Available</u>
	<u>Tape Drives:</u>	<u>No. of Channels</u> <u>1401-Compatible?</u>
<u>Program Requirements:</u>	<u>Storage</u>	<u>Packed Word Size</u>
	<u>Tape Drives</u>	<u>Links?</u>
	<u>Dependence on Assembly Language</u>	

Planned Program Usage:

Planned Program Changes:

Program Innovations We are Interested in:

- 1) IBM 360 version.
- 2) Version with ENDF/B cross sections.

Program Problems/Errors:

Program Changes of Immediate Interest:

PARTICIPANT INTRODUCTION FORM

Agency: BRL, Vulnerability Laboratory, Aberdeen Proving Ground, Md. 21005

Name(s) of Representative(s): M. J. Reisinger

Program Used: MAGIC

Purpose: Currently being used to debug target descriptions for Electronics Command, Army Tank Automotive Command, Missile Command, Munitions Command, Weapons Command, Nuclear Effects Laboratory, Test & Evaluation Command and Falcon Research & Development.

Computer(s) Used: Name BRLESC Word Size 68 bits

Memory Size: Total 120K Available 80K

Tape Drives: No. 3 No. of Channels 7/9 1401-Compatible? Yes

Program Requirements: Storage 48 Packed Word Size 30

Tape Drives 4 Links? No

Dependence on Assembly Language Depends on version

Planned Program Usage:

- 1) "Graphic Program" being developed from NASA program by L. Bain and M. J. Reisinger.
- 2) Recognition of heat projectile improper detonation from target description.
- 3) Point burst program with emphasis on components.

Planned Program Changes: Generalized Ellipsoid (i.e., not restricted to ellipsoids of revolution). Elimination of enter-leave table philosophy in favor of a more direct approach which is intended to reduce tracking time.

Program Innovations: Graphics Package.

Program Problems/Errors: A more detailed description of targets than done in the past (example, M60A1 with approximately 2500 bodies) is rapidly approaching our computer system time and size limit: a 1300 body description is using 64K, will 2500 bodies use less than the available 80K? Computer time on our system forces partial runs for graphics (need about four hours, large memory). Summation: need better computer.

Program Changes of Immediate Interest:

- 1) Development of support subroutines such as generalized components (wheels, ammunition, engine, etc.) that require location, orientation, and relative size that lead to computer generated bodies (solids).
- 2) Development of programs that would construct the optimized solid for a body from an input consisting of point data read directly from engineering drawings.

3. RECENT ACTIVITY WITH MAGIC AND SAM-C

The three sections that follow consist of the material presented verbally at the conference plus one or two minor additions or modifications.

3.1 Recent Activity with MAGIC at AMSAA. (Presentation by Larry Bain)

The recent activity with MAGIC at AMSAA falls into one of two categories: program changes or proposed plans. Each category is discussed separately.

3.1.1 Changes to MAGIC. This category is divided into three subtopics:

- Additions.
- Modifications.
- Corrections.

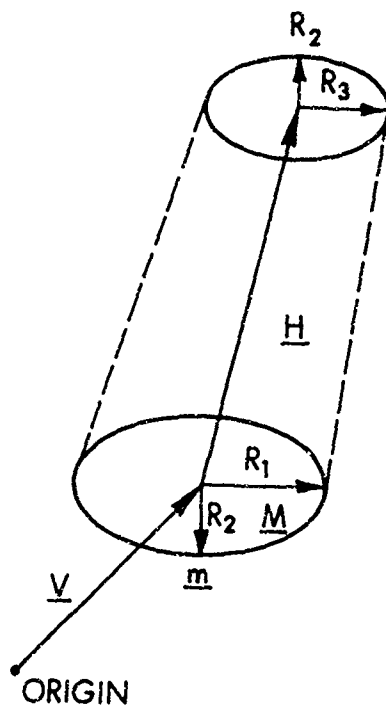
a) Additions. Of primary interest, three new solids have been added:

- TEC (Truncated Elliptic Cone).
 - i) Height vector does not need to be perpendicular to the base ellipse.
 - ii) Specify (Table 3.1)
 - V - vertex of base ellipse,
 - H - height vector,
 - M - semimajor axis of base ellipse,
 - m - semiminor axis of base ellipse, and
 - R - ratio (base ellipse/top ellipse),
viz., $R = (R1/R3) = (R2/R4)$.

(The normal height vector is computed in GENI; $\underline{n} = \underline{M} \times \underline{m}$; it is necessary to change the sign of \underline{n} if $\underline{H} \cdot \underline{n} < 0$.)

TABLE 3.1 CARD INPUT FOR THE NEW SOLIDS

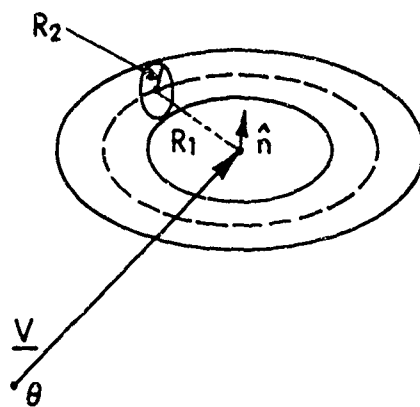
Body Type	3-Letter ID		No. of Cards Needed				
	4 - 6	11-20	21-30	31-40	41-50	51-60	61-70
Truncated Elliptic Cone	TEC	V _x M _x [*] R	V _y M _y [*]	V _z M _z [*]	H _x m _x [*]	H _y m _y [*]	H _z m _z [*]
Torus	TOR	V _x R1	V _y R2	V _z	N _x	N _y	N _z
Arbitrary Curved Surface	ARS	M X(1,1) . . X(1,N) X(2,1) . . X(M,1) . .	N Y(1,1) Y(1,N)	Z(1,1) Z(1,N)	X(1,2) Z(1,2)	Y(1,2) Z(1,2)	3 of n N+2 of n
* \underline{M} → semimajor axis \underline{m} → semiminor axis		X(M,N)	Y(M,N)	Z(M,N)			



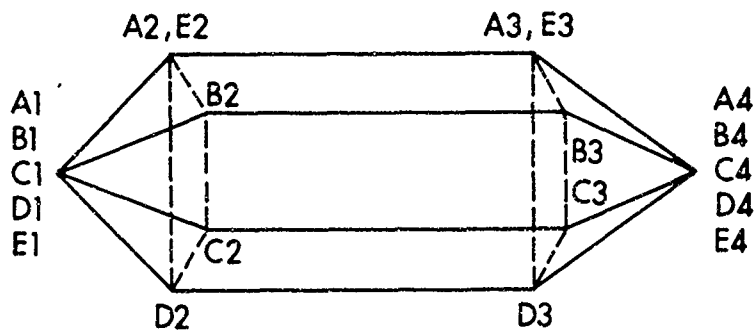
• TOR (Torus)

Specify (Table 3.1)

- \underline{V} - vertex,
- \underline{n} - normal vector (normal to the plane bisecting the torus),
- R_1 - radius from \underline{V} to the mid-point of the torus' cross section, and
- R_2 - cross sectional radius ($R_1 \geq R_2$).



- ARS (Arbitrary Surface)
 - i) Specify M curves of N points (see Table 3.1).
 - ii) The number of words of memory is $92 + M*N*4$.
 - iii) An ARS can be described in more than one way.



Example (using the figure above).

One Approach (Solid Lines)		Another Approach (Dashed Lines)	
M = 5	N = 4	M = 4	N = 5
curve 1	pt A1, A2, A3, A4	curve 1	pt A1, B1, C1, D1, E1
curve 2	pt B1, B2, B3, B4	curve 2	pt A2, B2, C2, D2, E2
curve 3	pt C1, C2, C3, C4	curve 3	pt A3, B3, C3, D3, E3
curve 4	pt D1, D2, D3, D4	curve 4	pt A4, B4, C4, D4, E4
curve 5	pt E1, E2, E3, E4		
Figure closed by duplicate curve		Figure closed by duplicate points	
Note: pt A1 = B1 = C1 = D1 = E1		pt A2 = E2	
pt A4 = B4 = C4 = D4 = E4		pt A3 = E3	

The addition of these new solids has required the addition of auxiliary subroutines:

- QRTIC to solve 4th degree equation.
- CUBIC to aid QRTIC.

- CROSS to compute vector cross products.
- DOT to compute dot products.
- UNIT to compute unit vectors.
- ARIN to process the ARS input.

Additional coding is also similarly required in subroutine CALC to compute normals and in GENI, G1, and WOWI.

Quite apart from the new solids, coding has been incorporated in Geni to increase input checking as follows:

- Checks of vector perpendicularity in BOX, RAW, REC, and TEC.
- Checks of TRC radii to ensure that $R_B \neq R_T$.
- Checks of TOR radii to ensure that R_1 is not less than R_2 (v.s.).

Schematically, an option has been added to suppress tape 8 (the monitor output) output (except for error messages) when writing tape 1.

Finally, two control subroutines have been created to assess quantities other than line-of-sight thickness:

- AREA to compute presented areas (the ray is traced to its first contact).
- MOMENT to compute moments of inertia (and as a by-product, the center of gravity, total weight, total volume, mean angle of incidence, and the mean cosine of incidence are also computed).

b) Modifications. Seven modifications have been made:

- The ELL input has been optimized.
 - i) Present input (cc 7-10 = 0 on card 1) is both foci plus the length of the major axis.
 - ii) Optimal input (cc 7-10 \neq 0 on card 1) requires the vertex, a vector representing the semimajor axis, and the scalar length of the semiminor axis.
- Computer word packing has been converted from 45 bits/word to 30 bits/word (it is estimated that 30 bit packing runs about 30 percent faster on BRLSEC I and II).

- The solid input section has been revised by eliminating FLOCON and DIGCON and substituting the F-type format specification.
- SENSESWITCH settings have been eliminated and their control data read in on punch-cards.
- The data output coding in TRACK has been revised to eliminate SETUP and ISIGN by using I and F-type format specifications.
- A version of MAGIC has been written in "standard" FORTRAN (incidentally, this version runs slower on the BRLESCs than any of the versions already mentioned).
- The control logic in the main program has been modified so that VOLUM can be run without reading in the Identification Table.

c) Corrections. Four major subroutines were found to have more or less subtle errors:

- VOLUM faltered when G1 tried to combine regions of the same "item" code because VOLUM requires that each region be processed separately; a special exit was added to G1 to correct this condition.
- TESTG suffered a similar fate but to a greater extent since the item data was not loaded into core prior to the execution of TESTG; an additional special exit was added to G1 to rectify this condition.
- GENI computes data for the normal vector to the base ellipse in the TEC but failed to ensure that the normal was an inward rather than an outward normal; a check was added so that if $H \cdot n$ is negative the direction of n is reversed.
- CALC was unable to correctly calculate normal thickness through adjacent regions with the same item code (cf., VOLUM and TESTG); a modification to the existing exit in G1 to compare the item code of the next region with the item code of the previous region was made to allow continuing the normal ray.

3.1.2 Proposed Plans for MAGIC. Activity in four areas is being carried on:

- Compatibility with SAM-C - let MAGIC do some of the geometry processing for SAM-C.
- Eliminate part or all computer word packing.

- Couple the target description ray tracing directly to outputs such as vulnerable area, kill probabilities, etc.
- Addition of a graphics capability via plotters, line-printers, CRT displays, or all three.

3.1.3 Concluding Remarks. The actual changes required to implement the three new solids are discussed in the MAGI report, A Description of Three Additional Bodies for the MAGIC Conventional Vulnerability Program, by J. R. Davis and M. Moskowitz (MAGI report MR 6902, May 1969). An abridged version of this report consisting of the main portion of the text appears as Appendix B.

In any direct dealings with the coding of the MAGIC program, a knowledge of the core storage layout, input data requirements, etc., is essential. Figures 3.1 through 3.4 supply the requisite data:

- Figure 3.1 is a map of the MASTER/ASTER array showing storage for the processed geometry data. Both 45-bit and 30-bit packing versions are shown. Variable names beginning with L are the locations of each set of data in the MASTER/ASTER array.
- Figure 3.2 concludes the map of the MASTER/ASTER array showing storage of the identification table and the "working" storage used at run time.
- Figure 3.3 is a map of the pointers to the location of the solid data.
- Figure 3.4 is the map of the storage for the arbitrary surface (ARS).

Finally, to appreciate the relationships between various routines, Figure 3.5 displays the many auxiliary routines of MAGIC and their relationship to the main of "driver" routines.

3.1.4 Additional Information. In October, 1969, several runs were made using the AMSAA "Revised Standard MAGIC" (Appendix D) on a number of different computing systems. The geometry input consisted of a description which we shall call the "December '68 Master Target." This target is comprised of 701 solids and 904 regions; none of the three new solids were used. About 5 man-months were required to create the description. The following driver routines and their input were used:

GRID	0° Az 0° Elev	1015 cells
VOLUM	Head-on	1015 cells
AREA	Head-on	4189 cells
TESTG	-----	2 rays

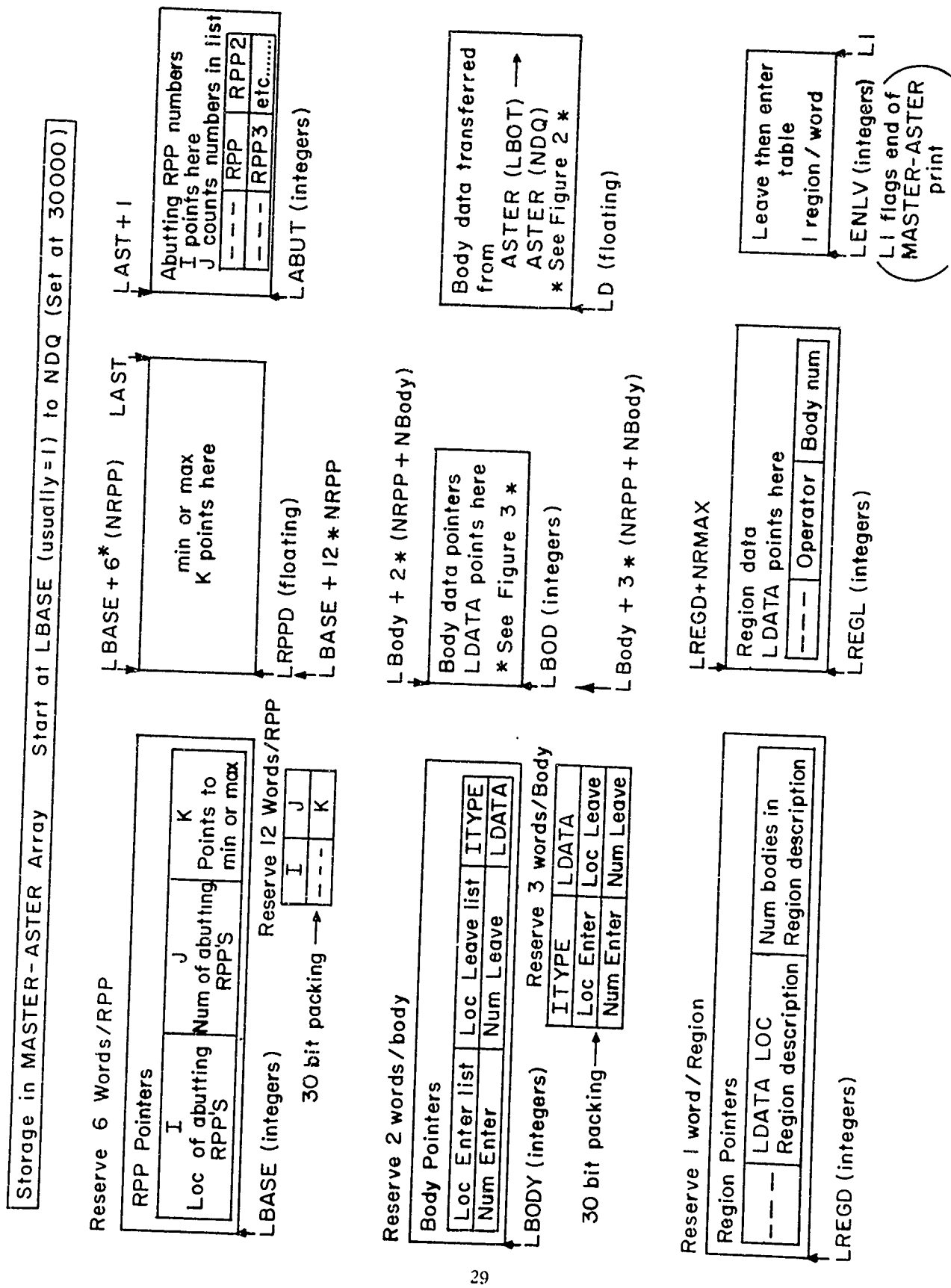
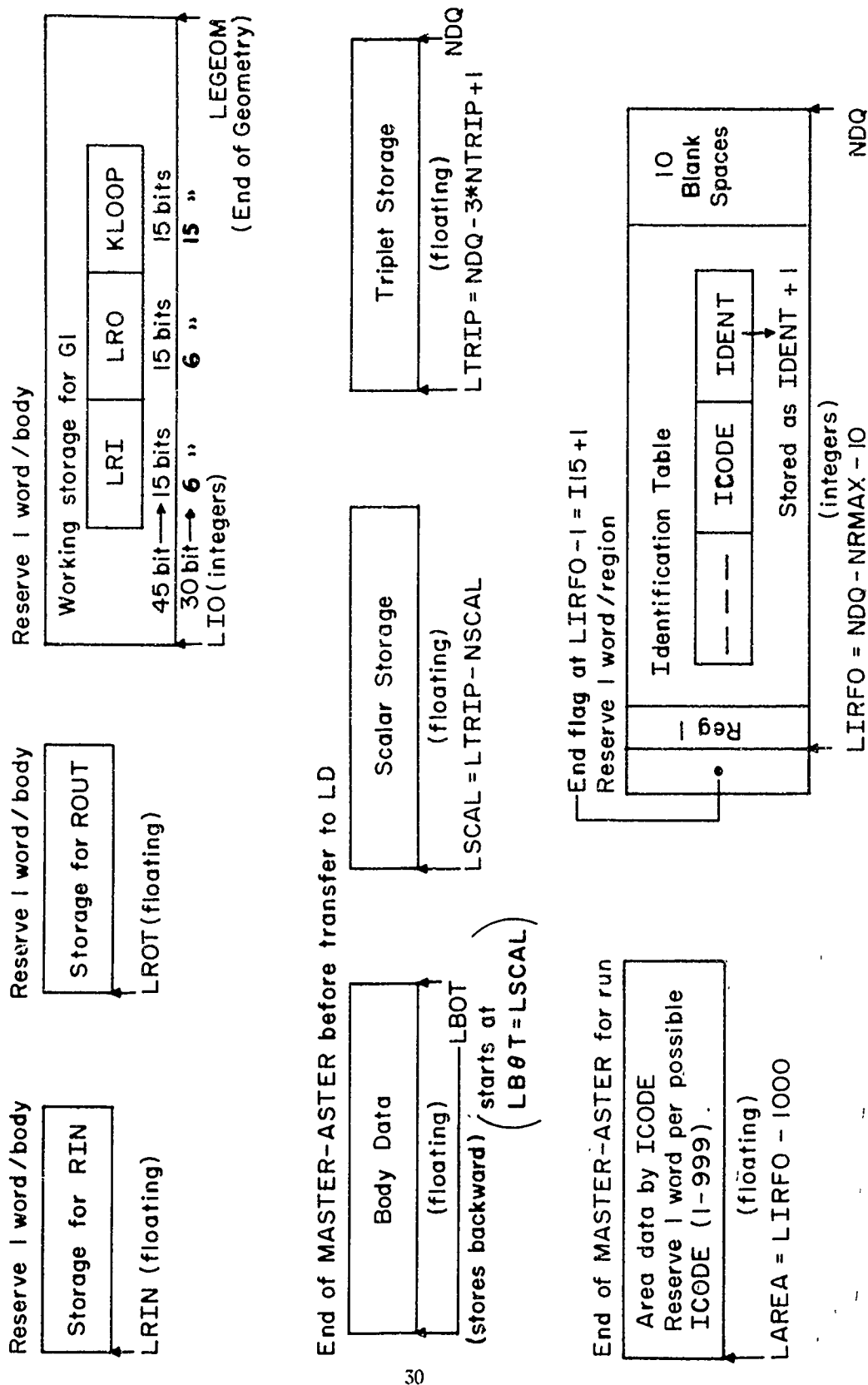


Figure 3.1 Partial Map of the MASTER-ASTER Array



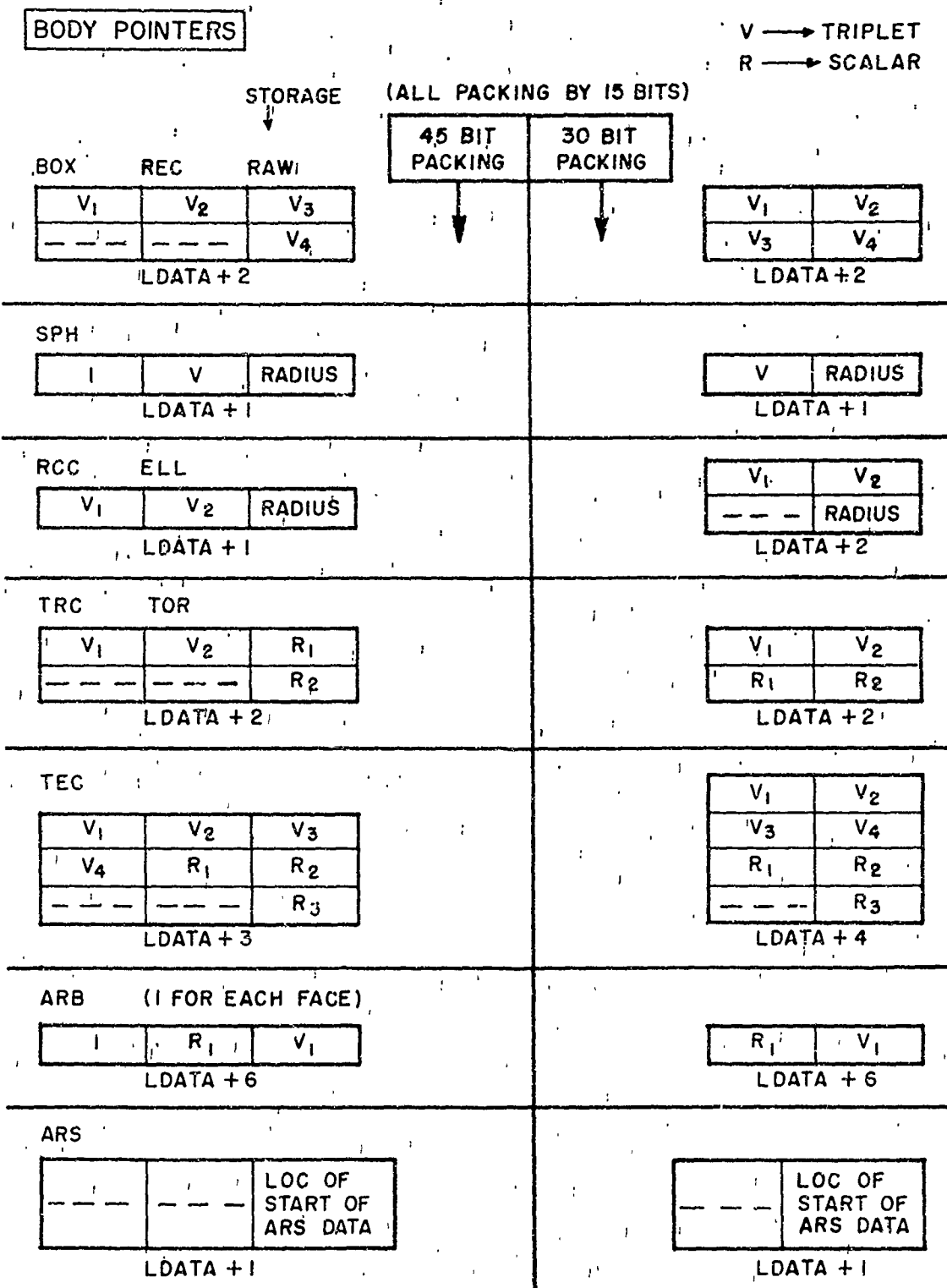
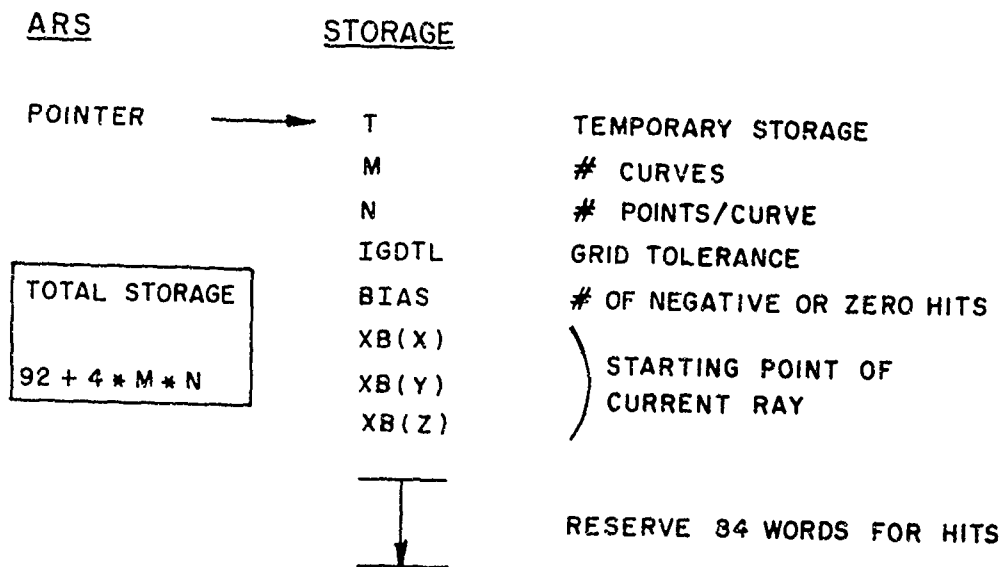


Figure 3.3 Map of Pointers to Solid Data



Rest of STORAGE M Sets of N Points

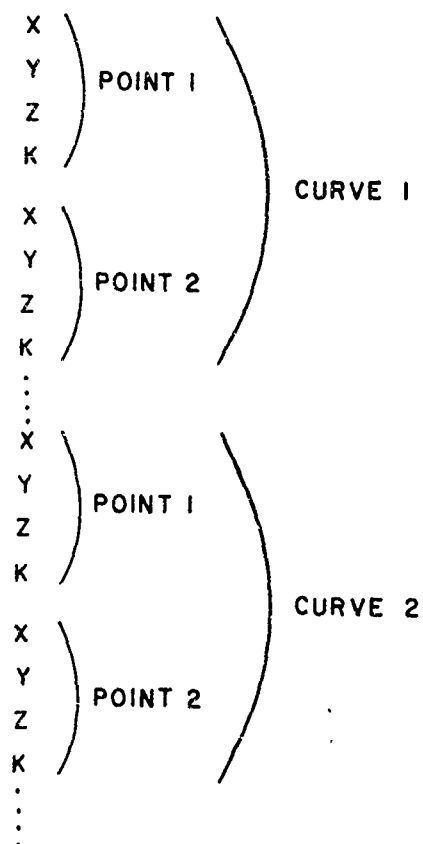


Figure 3.4 Storage for the ARS

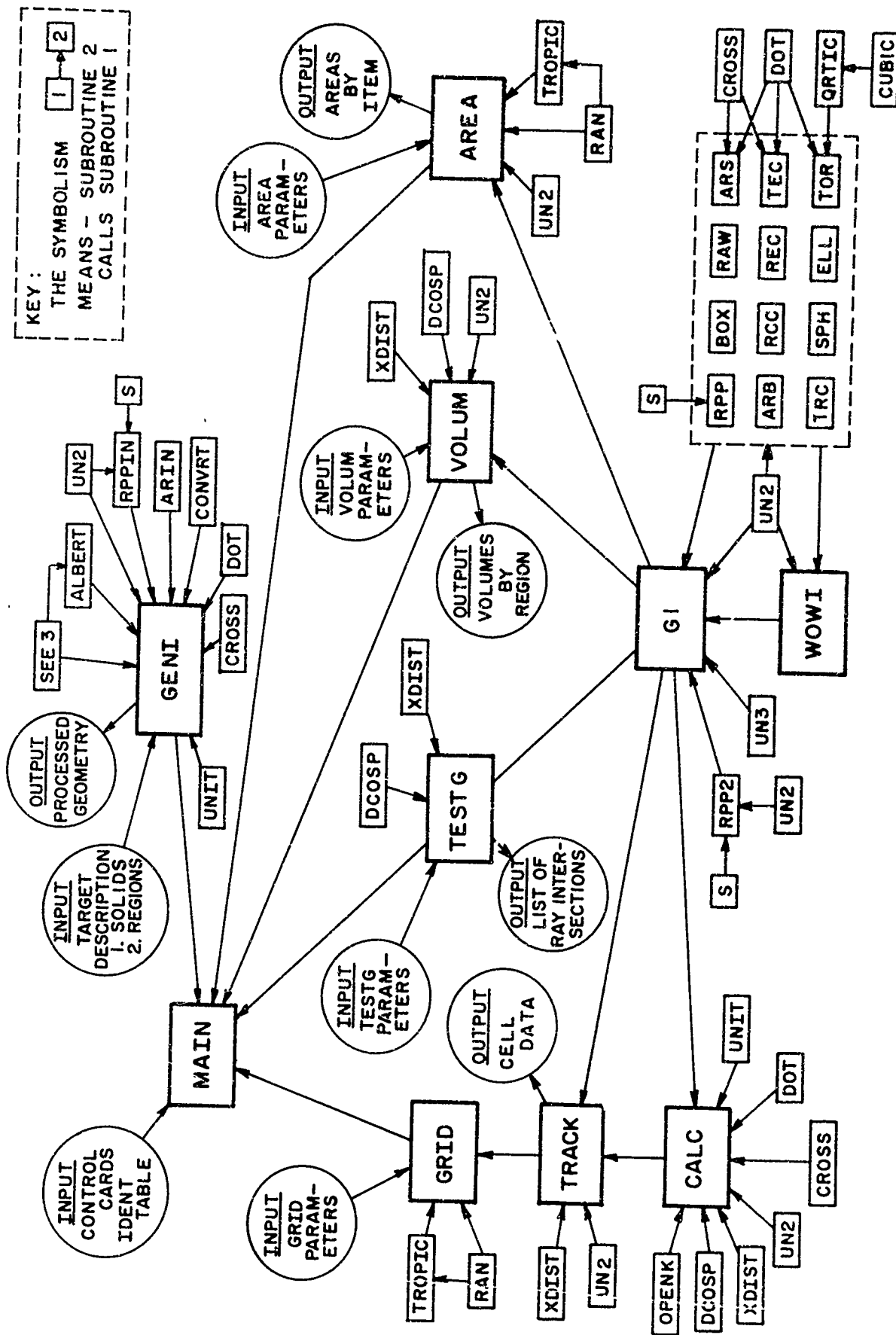


Figure 3.5 Interrelationships of MAGIC Routines

Table 3.2 presents the results of the various computer runs.

TABLE 3.2 COMPUTER SYSTEM TIME REQUIREMENTS

System	Location	Time (in minutes)	Ratio
BRLESC I	ARDC, APG, Md	133	1
BRLESC II	ARDC, APG, Md	55*	.41
CDC 6600	New York Univ	15	.11

* Using the on-line printing capability instead of "dumping" onto tape for off-line tabulation.

3.2 Recent Activity with SAM-C at NEL. (Presentation by Wayne Coleman)

The recent activity with SAM-C at NEL falls into one of three categories: recent calculations completed using SAM-C, completed corrections and improvements to the SAM-C code, and plans for the future.

3.2.1 Recent Calculations. This category is divided into three subtopics:

- Calculation of the energy dependent gamma flux at 3 feet above an infinite Co-60 "fallout" field.
- Calculation of the energy dependent and total neutron fluence at various positions within the Ralph J. Truex (Tandem Van de Graff) accelerator at the Nuclear Effects Laboratory (Edgewood Arsenal).
- Calculation of the energy dependent neutron fluence in an environment simulating that used in Operation HENRE.

a) Infinite Fallout. Although these calculations did not constitute a comprehensive test of the geometric capabilities of SAM-C, very good agreement with the known solution of this problem was obtained.

b) Accelerator. These calculations include the most complex geometries that have been simulated to date at NEL and were included for that reason. The physical results unfortunately cannot be compared directly to any results of experiments or any other calculations.

c) HENRE. This problem demonstrates how, under certain restrictions, ad hoc changes can be made to produce results for source angular distributions other than 4π -isotropic or monodirectional. The results compared favorably with unfolded flux spectra from experimental activation measurements when the SAM-C results were used as an "input guess spectrum" in the unfolding calculations.

3.2.2 Corrections/Improvements. Corrections are indicated by "C" while changes that are more in the category of improvements are indicated by "CI." C/CI's are by order of their appearance in the program.

a) Variable Dimensioning (CI). On machines that assign priority based on memory requirements, adjustable sizes of blank common and labeled common CROSA are an operational necessity. Variable dimensioning in SAM-C is accomplished by creating an artificial "main" program in which dimensioning is accomplished.

b) SEEK (C). Change $E = 1$ to $I = 1$ in TUNC and MONTE.

c) SOUCAL (C). Set "KKMAX = K-1" between FORTRAN statement numbers (S.N.) 780 and 925.

d) CARLO (C). The calculation of flux-at-a-point following an inelastic collision is incorrect. See the August RSIC Newsletter (No. 57) for details.

e) SOUPIC (C/CI). Volume source additions may be made by modifying the coding shortly after FORTRAN S.N. 800.

f) ARB (C). T was inadvertently used as a variable name to represent time in a labeled common and temporary storage in the routine (S.N. 50 and $50 + 1$).

g) FAP (CI). The coding for identifying point flux contributions by region appears here.

h) FAP (C). To correctly calculate flux-at-a-point add COMMON/LSU/LSURF.

i) GE (C). (1) The scattering problem look-up for neutrons was referencing the wrong area of common for neutron anisotropic scattering; cf., the RSIC version. (2) The third argument in the call to SEEK (S.N. 185) should be 10 and not 11. (3) Format 106 ends incorrectly - replace "10" with any desired E- or F-type format.

j) SUBED (C). Change "NRMAX" to "NDET" at S.N. 121.

k) SOUCAL (C). Change Format 402 (not all versions). (CI) finally changes to compute statistics on the total flux or fluence for flux-at-a-point would be desirable.

3.2.3 Improvements in Progress. Modifications to calculate flux at a point as a function of angle or direction as well as energy are well underway at NEL.

3.2.4 Planned Improvements. Neutron cross section representation is earmarked for extensive study. Key inputs or approaches to this subject include UNC-SAM-3, an ENDF/B cross-section processor, and a thorough examination of the representation of neutron elastic, angular scattering distributions. Total, elastic, and non-elastic cross section data are also expected to be involved.

3.3 Coordination of MAGIC and SAM-C.

Both MAGIC and SAM-C process card-image target description data and store the results in the MASTER/ASTER array. Both can compute volumes. If the conventional component/space code table has been created, a trivial program (Program COMPAS) exists to convert this data to a region/chemical composition assignment table. The following two sections describe the changes required so that target descriptions can be utilized by either MAGIC or SAM on a wide variety of computational equipment with a minimum of difficulty. The goal, of course, is to permit the creation of a library of target descriptions which can serve two purposes: they can reduce duplication of effort and they can lend insight into what a given agency considers an adequate degree of descriptive detail.

3.3.1 Discussion. This category is divided into two subtopics: a description of the proposed tape's contents and a discussion of how the tape should be created.

a) Description of the Tape Contents. The first step is to identify the data available for the library tape; these data are displayed in Table 3.3.

TABLE 3.3 TARGET DESCRIPTION DATA AVAILABLE

MAGIC	SAM-C
"Processed" (GENI) Target	Output Data (MASTER-ASTER*)
Region Identification Table	Composition Assignment Table
Volumes (Optional)	Chemical Composition Definitions
Region Weights (Optional)	"Processed" (BAND/BEDIT) Cross Section Data (MASTER-ASTER)
Moment of Inertia (Optional)	Volumes (ASTER)
Target Description Title	
"Raw" (Card-Image) Target Description Solid & Region Data	

*The SAM-C program stores these data in a location different from that used by MAGIC.

TABLE 3.4 LIBRARY TAPE CONTENTS: PARTS 1 AND 2

"Block" 1. Identification (80A1)

Target Description Title

"Block" 2. Table of Contents

(1) Solid/Region Table Format (cc 1-5)

0 : GENI Input form

≠ 0 : GENI Output form

For (2) through (8), 0 means the category was not included while a non-zero entry indicates that the data was included

(2) Region Identification Table (cc 6 - 10)

(3) Composition Assignment Table (cc 11 - 15)

(4) Volumes (cc 16 - 20)

(5) Chemical Composition Data (cc 21 - 25)

(6) Region Total Weights (cc 26 - 30)

(7) "Organized" Cross-Section Data (cc 31 - 35)

(8) Moment of Inertia Data (cc 36 - 40)

(9)-(10) (Spares) (cc 41 - 50)

cc 51-52 Unit Systems (Examples: IP for inches, pounds (& seconds),
CG for centimeters, grams, seconds, etc.)

cc 53 Coordinate System "Handedness" L = left, R = right

cc 54-80 Location of Geometric Origin with respect to the Reference
Origin (e.g., for "tanks" the reference origin is frequently
the intersection of the turret datum line and the vehicle
centerline) (3E9.2)

The second step is to determine the data to be loaded and in what order. Clearly, the target description title plus some type of flagging to indicate the data categories available should appear near the start of the tape to minimize the time spent in identifying the tape contents. It is therefore proposed that the first tape "block" consist of ten flag words as described in Table 3.4. Beyond these two points, any data that describe the target are suitable for inclusion on the tape.

b) Approach. It was the consensus of the program users that, while all library tapes should be in BCD format for transmittal, the basic target description solid and region data should be in card-image rather than in "processed" (GENI output) form; except for a few installations, the computer time spent in reprocessing the "raw" data is negligible compared with the time that would be used to convert the data into the form required by those of a different installation. It was also agreed that the raw data approach would place the least number of restrictions on the internal operating procedures of any given MAGIC or SAM-C program user.

3.3.2 Program Changes. Some coding changes of a very minor nature will be required. The routines affected are:

a) MAGIC. The main program plus GENI, VOLUM, the moment of inertia, and the vulnerable area routines will require modification.

b) SAM-C. TUNC, GENI, DR, and VOLUM are involved. In SAM-C, VOLUM will generally only require an additional output statement.

4. DEFINITION OF MAGIC AND SAM-C; CREATION OF SOURCE DECKS

This provided the first opportunity of the conference for the conferees to determine the course of action to be followed. The two sections that follow present the consensus achieved by the conferees in defining what the capabilities of the two programs should be and in determining how these capabilities should be achieved.

4.1 MAGIC.

The following sections represent the major areas discussed at the conference together with the results of these discussions.

4.1.1 Standard Version. A consensus was achieved in four areas:

- a) Input. The input to MAGIC will consist of
- RPP Data,
 - Solid Description Data, and
 - Region Description Data.

b) Subprograms. The subprograms are characterized as "geometry processing" (including some testing):

- GENI,
- ALBERT,
- RPPIN, and
- ARIN,

or "ray tracking" (but not in the sense of GRID which is considered a "driver routine"):

- G1,
- WOWI/RPP2,
- RPP/body routines + TOR, ARS & TEC, and
- Auxiliary body routines such as QRTIC, UNIT, etc.

c) Program Features.

- No packing.
- It should be possible to use an RPP to subdivide the target, itself.
- Drop TESTG.
- Drop FLOCON and DIGCON.
- Output the processed geometry which should consist of titling data, the geometry data, and the functional identification table.

d) Tests. It was agreed that TESTG as a random but supposedly complete test of the description is inefficient and should be dropped; in its place, a driver routine of interest - such as GRID - should be used. Input testing was considered valid and at least the following tests should be available:

- Legitimacy of solid names.
- Vector perpendicularity for boxes, RAW's, and the REC.
- Equal radii in the TRC.
- Region checking (on an optional basis).
- "4-points-in-a-plane" in the ARB.
- Degenerate plane in the ARB.

- Proper ordering of RPP input.
- Proper structure creation by contiguous RPP's (since the structure that would enclose all RPP's must be in the shape of an RPP).

4.1.2 Ad Hoc Problems. Although there was not enough time for the formal formation of working groups to solve ad hoc problems, the following problems were defined for solution by any interested groups or individuals.

- The creation of a technique for the arbitrary designation of solids to have the special characteristics currently displayed by the RPP.
- The identification of methodological differences in the routines that form MAGIC between versions held by the several agencies using MAGIC.
- The establishment of a methodology for creating "library" routines (for such configurations as wheels, people, etc.) which can be processed as a unit rather than a set of distinct subsolids.
- The establishment of a uniform system of flags for transmittal of the processed geometry.

4.2 SAM-C.

SAM-C was not the subject of serious discussion until late in the afternoon of the second day. Because the SAM-C program is so large and complex, our attitude toward it is considerably different than our attitude toward MAGIC. In the first place (and of overriding importance), there are few computing facilities capable of efficiently executing the SAM-C programs; secondly, a substantial amount of understanding of the code and the manner in which it attempts to solve problems, and of the problems themselves, is required to achieve any sort of successful solution. Keeping these points in mind, the following sections present the consensus reached.

4.2.1 Standard Version. For the time being at least, the standard version of SAM-C will be that obtainable from RSIC at Oak Ridge. In brief, this version features Combinatorial Geometry inputs identical to those of Standard MAGIC with one important difference: triplet and scalar inputs will be allowed.

4.2.2 Ad Hoc Problems. Although time was again insufficient, the following study/problem areas were defined:

- The abolishment of computer word packing to the greatest practical extent.

- The incorporation of ENDF/B cross sections - preferably using a noncommon energy mesh.
- Bonafide time dependence: time dependence currently assumes the source is a separable function of time.
- Simplification of input preparation.

5. CREATION OF BENCHMARK PROBLEMS

Although time constraints precluded the actual creation of test problem input, it was possible to indicate what the benchmark problems should include.

5.1 MAGIC.

The test problem input should meet five conditions:

- Use only the solids generally available; place the three new solids at the end of the solid table for possible deletion.
- Use all three region operator symbols (+, -, OR).
- Use 1 RPP to enclose the target (Wright-Patterson AFB will use a BOX).
- Employ solids in such a way that they present overlaps and contiguities.
- Similarly, ensure that at least one situation arises where more than one following region has the same functional identification code as the region in front (to ensure that the correct normal thickness is being computed).

In addition, it is desirable to introduce a few deliberate errors to ensure that internal error checks are operative.

5.2 SAM-C.

It was decided that more than one benchmark problem should be created to enable checking the modeling of the physical solution and the execution of the code. Two benchmark problems were agreed upon:

- The "infinite" fallout problem (RSIC Benchmark Problem No. 4), and
- AFWL "Rocket" Geometry.

Physical/code input for these problems is presented in Appendix C.

Next page is blank.

APPENDIX A. MAGIC/SAM-C CONFERENCE PARTICIPANTS

<u>Name</u>	<u>Agency</u>	<u>City, State</u>
1. Robert E. Barnas	Picatinny Arsenal	Dover, New Jersey
2. Sue Gibson	AFATL	Eglin AFB, Florida
3. Larry Bain	AMSAA (Methodology)	APG, Md.
4. Matthew J. Reisinger	BRL (VL)	APG, Md.
5. John A. Zook	AMSAA (ASA)	APG, Md.
6. Thomas Jankunis	Picatinny Arsenal	Dover, New Jersey
7. Jesse W. Brewer	AMSAA (ASA)	APG, Md.
8. Donald W. Mowrer	BRL (VL)	APG, Md.
9. Robert E. Walther	BRL (VL)	APG, Md.
10. Lewis G. Gotze	BRL (VL)	APG, Md.
11. John Saarmann	Picatinny Arsenal	Dover, New Jersey
12. Michael J. Paul	AFWL (WLRAS)	Albuquerque, N. M.
13. A. Kris Widdison	AFWL (WLDC-PD)	Albuquerque, N. M.
14. Kenneth Richer	BRI. (S&PL)	APG, Md.
15. George H. Connor, Jr.	BRL (NEL)	Edgewood Arsenal, Md.
16. Robert W. Roussin	ORNL - RSIC	Oak Ridge, Tennessee
17. Wayne A. Coleman	BRL (NEL)	Edgewood Arsenal, Md.
18. W. B. Beverly	BRL (NEL)	Edgewood Arsenal, Md.
19. Janet Lacetera	BRL (NEL)	Edgewood Arsenal, Md.
20. Richard Saum	BRL (NEL)	Edgewood Arsenal, Md.
21. William Ralph	NWL	Dahlgren, Virginia
22. Robert E. Gray	NWL	Dahlgren, Virginia
23. Roy R. Hilbrand	ASD	Dayton, Ohio
24. Gerald Bennett	ASD	Wright-Patterson AFB, Ohio
25. Robert Kesselman	Picatinny Arsenal	Dover, New Jersey
26. Joe Burgio	Picatinny Arsenal	Dover, New Jersey
27. Norman S. Banks	BRL (TBL)	APG, Md.
28. Robert Lake	AMSAA (ASA)	APG, Md.
29. Ronald Marking	AMSAA (ASA)	APG, Md.

APPENDIX, B MAGI REPORT MR 6902 (ABRIDGED)

CONTENTS

	Page
B.1 ORGANIZATION OF THE NEW MAGI PROGRAM	47
B.1.1 Changes to CALC	47
B.1.2 Changes to GENI	49
B.1.3 Changes to G1	49
B.1.4 Changes to WOWI	49
B.1.5 Addition of TEC	49
B.1.6 Addition of TOR	49
B.1.7 Addition of QRTIC	49
B.1.8 Addition of CUBIC	50
B.1.9 Addition of DOT	50
B.1.10 Addition of CROSS	50
B.1.11 Addition of UNIT	50
B.1.12 Addition of ARS	50
B.2 DESCRIPTION OF INPUT PARAMETERS	52
B.2.1 Truncated Elliptic Cone	52
B.2.2 Torus	52
B.2.3 Arbitrary Surface	52
B.3 BODY CARDS	55
B.4 NEW VARIABLES IN COMMON	55
B.5 NEW ERROR STOPS	56

Next page is blank.

B.1 ORGANIZATION OF THE NEW MAGIC PROGRAM

The general organization of the MAGIC program remains as before. The following subroutines were changed: CALC, GENI, G1, WOWI. The following routines were added: TEC, TOR, ARS, DOT, CROSS, UNIT, QRTIC, CUBIC, ARIN. A description of the changes to existing subroutines and the new subroutines follows:

B.1.1 Changes to CALC.

CALC - Statement 18 - Change the computed "go to" by adding locations for computing the normals in the new bodies.

Statement 500

This section computes the normals to the elliptic cone. If the point of intersection was the top or bottom the normal is the same as the normal received as input. If the point is on the side of the cone the following equation is used.

$$(1) \quad \bar{W}_B = \frac{(\)_1}{\tau} \bar{A}^* - \frac{\bar{H} \cdot \bar{A}^*}{\bar{H} \cdot \bar{\eta}} \bar{\eta} + \frac{(\)_2}{1} \bar{K}^* - \frac{\bar{H} \cdot \bar{K}^*}{\bar{H} \cdot \bar{\eta}} \bar{\eta} - m(r_4 - r_2) \frac{\bar{\eta}}{\bar{H} \cdot \bar{\eta}},$$

$$\text{where} \quad (\)_1 = (\bar{X} - \frac{\bar{X} \cdot \bar{\eta}}{\bar{H} \cdot \bar{\eta}} \bar{H} - \bar{V} + \frac{\bar{V} \cdot \bar{\eta}}{\bar{H} \cdot \bar{\eta}} \bar{H}) \cdot \bar{A}^*, \text{ and}$$

$$(\)_2 = (\)_1 \text{ with } \bar{K}^* \text{ instead of } \bar{A}^*$$

The terms are defined as follows:

\bar{H} = height vector of cone,

\bar{A}^* = unit vector - direction of major axis,

\bar{K}^* = unit vector - direction of minor axis,

τ = ratio of major to minor axis squared,

$$m = \frac{(\bar{X} - \bar{V}) \cdot \bar{\eta}}{\bar{H} \cdot \bar{\eta}} (r_4 - r_2) + r_2$$

\bar{X} = point of intersection,

$\bar{\eta}$ = normal to cutting plane, and

\bar{V} = center of base ellipse.

The derivation is shown on the following page.

The equation for the ellipse parallel to the base ellipse and through the point of intersection \bar{X} is

$$(2) \quad \frac{((\bar{X}-\bar{V}-\gamma\bar{H}) \cdot \bar{A}^*)^2}{\tau m^2} + \frac{((\bar{X}-\bar{V}-\gamma\bar{H}) \cdot \bar{K}^*)^2}{m^2} = 1,$$

$$(3) \quad \gamma = \frac{(\bar{X}-\bar{V}) \cdot \bar{n}}{\bar{H} \cdot \bar{n}}, \text{ and}$$

$$(4) \quad m = \gamma r_4 + (1-\gamma)r_2.$$

NOTE: γ and τ and \bar{X} are known.

Expanding we get

$$0 = \left(\frac{\bar{X} \cdot \bar{n}}{\bar{H} \cdot \bar{n}} \frac{\bar{H} \cdot \bar{V}}{\bar{H} \cdot \bar{n}} \frac{\bar{V} \cdot \bar{n}}{\bar{H} \cdot \bar{n}} \bar{H} \right)^2 \cdot \bar{A}^* + \left(\frac{\bar{X} \cdot \bar{n}}{\bar{H} \cdot \bar{n}} \frac{\bar{H} \cdot \bar{V}}{\bar{H} \cdot \bar{n}} \frac{\bar{V} \cdot \bar{n}}{\bar{H} \cdot \bar{n}} \bar{H} \right)^2 \cdot \bar{K}^* - m^2.$$

Differentiating, with respect to X , Y , and Z , and taking the unit vector of the result gives us Equation (1).

Statement 550

The section computes the normal to the torus at the point of intersection. The equation used is:

$$(1) \quad \bar{W}_B = \frac{\bar{X} - (\bar{C} + r_1 \bar{d}^*)}{r_2}$$

where \bar{X} is point of intersection,

\bar{C} is center of torus,

r_1 is distance from center to the locus of mid-point of circular cross section

$\bar{d}^* = \text{unit } \{\bar{d}\},$

$\bar{d} = \{\bar{n} \times (\bar{X} - \bar{C})\} \times \bar{n} = \text{direction of } r_1 \text{ in plane, and}$

r_2 is radius of circular cross section.

Statement 575

This coding examines the intersection with the arbitrary surface (XI), selects the proper normal from those left in the MASTER/ASTER array by ARS, and places this normal into WB. See write-up of ARS routine to determine details of computation.

B.1.2 Changes to GENI.

GENI - Change all computed GO TO's involving body type to add three new bodies. Add conversion routines to store TORUS and Elliptic Cone data in either floating point or triplet and scalar form. Add coding to store Arbitrary Surface in floating point format. (Subroutine ARIN).

B.1.3 Changes to G1.

G1 - For a new ray, processing remains the same, except for changing the computed GO TO to check the new bodies. For continuation of a ray, a check is made for TORUS or ARBITRARY SURFACE. If the body is neither of these, the previously computed value is used. However, if the body type is one of these a check is made to see if the distance we have traveled is greater than ROUT. If it is not then we use the existing values for RIN and ROUT. Otherwise, we reenter the body routine and compute the next RIN/ROUT set (if any).

B.1.4 Changes to WOWI.

WOWI - The same changes as were made to G1 apply here. A description of the new routines follow.

B.1.5 Addition of TEC.

TEC - Body routine for truncated elliptic cone. Computes RIN, ROUT, LRI, LRO for intersection of ray and cone. Uses DOT, CROSS, and SQRT.

B.1.6 Addition of TOR.

TOR - Body routine for torus. Computes RIN and ROUT; LRI and LRO are 1. If four roots are found it selects the first pair greater than DIST as RIN and ROUT. Uses QRTIC and CUBIC.

B.1.7 Addition of QRTIC.

QRTIC (A,B,C) - Solves quartic polynomial equation with unit leading coefficient, $X^4 + C_1X^3 + C_2X^2 + C_3X + C_4 = 0$.

Method is from J. V. Uspensky, "Theory of Equations,"
pp 94-95. Used by TOR.

A = Array of coefficients

B = Array of roots

C = Number of real roots

B.1.8 Addition of CUBIC.

CUBIC (A,B,C) - Solves cubic polynomial equation with unit
leading coefficient, $X^3 + C_1X^2 + C_2X + C_3 = 0$.

Method is from J. V. Uspensky, "Theory of Equations,"
pp 84-93. Used by TOR.

A = Array of coefficients

B = Array of roots

C = Number of real roots

B.1.9 Addition of DOT.

DOT (A,B) - Computes the dot product of vectors \bar{A} and \bar{B} .

B.1.10 Addition of CROSS.

CROSS (A,B,C) - Computes the cross product of vectors \bar{B} and
 \bar{C} and stores result in vector \bar{A} .

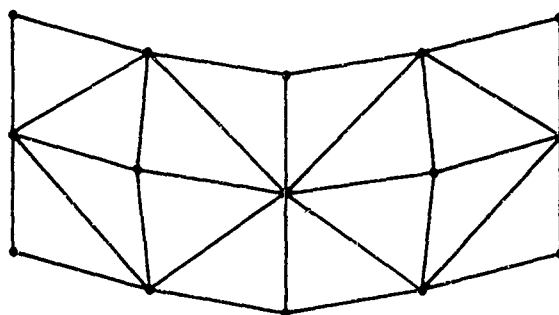
B.1.11 Addition of UNIT.

UNIT (A) - Computes unit vector of \bar{A} and stores back in \bar{A} .

B.1.12 Addition of ARS.

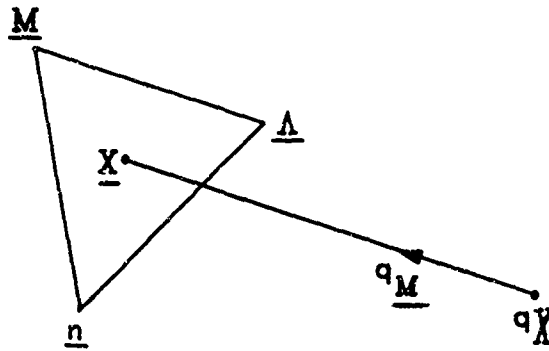
ARS - Body routine for arbitrary surface. Computes RIN, ROUT.
LRI and LRO are always 1.

For the purposes of determining intersections and normals, the
routine constructs a series of triangles, as below, and utilizes these
triangles to determine intersections and the associated normals.



When entered for a new ray, the intersections are stored in the MASTER/ASTER array, together with the normals at these points. Upon reentry for the same ray, the RIN, ROUT pair appropriate to DIST are selected and returned to the calling routine.

To determine the intersection and normals, the ray is tested against each triangle for which at least one point of the triangle lies within the projection of the grid square. The calculation is done as follows:



By the rules of convex figures in 2-space, for \bar{x} within the triangle, there must exist α, β, γ such that

$$(1) \quad \alpha \bar{u} + \beta \bar{v} + \gamma \bar{w} = \bar{x} = \bar{x}_b + s \bar{w}_b$$

$$\alpha + \beta + \gamma = 1; \alpha, \beta, \gamma > 0;$$

then

$$\gamma = 1 - \alpha - \beta,$$

$$(2) \quad \alpha \bar{u} + \beta \bar{v} + (1 - \alpha - \beta) \bar{w} = \bar{x}_b + s \bar{w}_b, \text{ and}$$

$$(3) \quad \alpha(\bar{u} - \bar{w}) + \beta(\bar{v} - \bar{w}) - s \bar{w}_b = \bar{x}_b - \bar{w}.$$

These are simply three equations in three unknowns.

Using determinants to solve this set of simultaneous equations we obtain α, β, γ , and s . After verifying that $\alpha + \beta + \gamma = 1$ and $\alpha, \beta, \gamma \geq 0$, we record the s value as well as a unit normal to the triangle. If α, β, γ fail to meet these requirements, the ray missed this triangle.

After performing the calculations for each triangle, the resulting s values and normals are placed in the MASTER/ASTER array. Sufficient space is provided for up to ten pairs of RIN and ROUT. The variable DIST is used to determine which pair of RIN and ROUT should be returned to the calling routine.

B.2 DESCRIPTION OF INPUT PARAMETERS

B.2.1 Truncated Elliptic Cone.

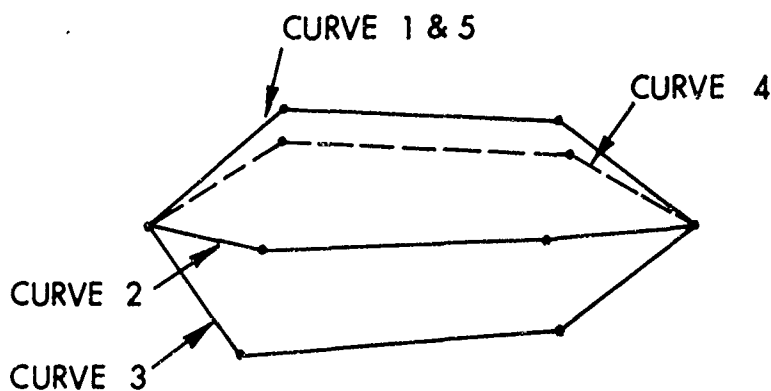
Specify a vertex V at the center of the larger ellipse, the height vector H , expressed in terms of its x,y,z components, the direction of the major axis A , the direction of the normal N , and three scalars - $R1$, the length of the major axis of the larger ellipse, $R2$, the length of the minor axis of the larger ellipse, and P , the ratio of the larger to the smaller ellipse.

B.2.2 Torus.

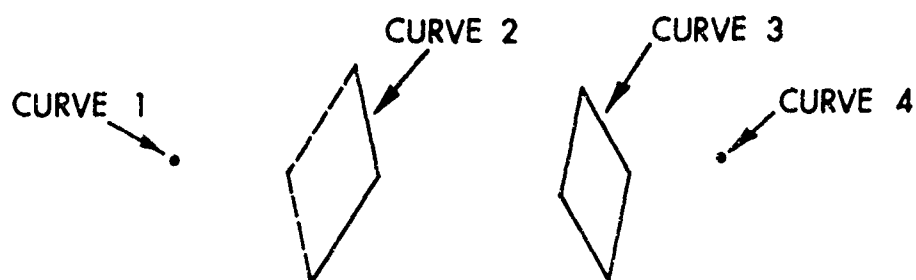
Specify a vertex V at the center of the torus, the normal to the plane in which the locus of mid-points of the circular cross sections lie, N , and two scalars - $R1$, the distance from the center to the mid-point of circular cross section, and $R2$, the radius of circular cross section.

B.2.3 Arbitrary Surface.

Specify the number of curves (M) to be specified and the number of points (N) which will be specified on each curve. A surface is constructed between curve 1 and curve 2, between curve 2 and curve 3, etc. The user must be careful that the described figure is closed and solid.



In the previous example, the first and last point of each curve is identical, and the first curve is identical to the last, and one can see that the figure is solid.



In this example, curve 1 consists of the same point repeated five times, curve 2 of five points (the first and fifth point being the same), curve 3 is defined similar to curve 2, and curve 4 similar to curve 1. It can also be seen that this figure is solid.

To further illustrate this figure, note in the figure that

$M = 4$	$N = 5$
Curve 1	pt. A_1
	B_1
	C_1
	D_1
	E_1
Curve 2	pt. A_2
	B_2
	C_2
	D_2
	E_2

M = 4

Curve 3

Curve 4

is identically the same figure as

M = 5

Curve 1

Curve 2

Curve 3

N = 5

pt. A_3

B_3

C_3

D_3

E_3

pt. A_4

B_4

C_4

D_4

E_4

N = 4

pt. A_1

A_2

A_3

A_4

pt. B_1

B_2

B_3

B_4

pt. C_1

C_2

C_3

C_4

M = 5	N = 4
Curve 4	pt. D ₁
	D ₂
	D ₃
	D ₄
Curve 5	pt. E ₁
	E ₂
	E ₃
	E ₄

The user should use this isomorphism as a check on whether the figure defined is truly closed and solid.

B.3 BODY CARDS

The data describing each body must be input using the format described in Table B.1. This table is similar to Table 3.1 (page 37 in the original document) and should be viewed as an extension of that table.

B.4 NEW VARIABLES IN COMMON

<u>Variable Name</u>	<u>Labeled Common</u>	<u>Definition</u>
IGRID	DAVIS	The grid square of the origin of the current ray (XBS)
LOOP	DAVIS	Set by G1 upon entry to a body routine to reflect ray number of last ray fired at the body.
INORM	DAVIS	Set by G1 to indicate if the ray is being fired normal to a surface (normal is computed by CALC)

B.5 NEW ERROR STOPS

<u>Routine</u>	<u>Message</u>	<u>Explanation</u>
CALC	ARS DID NOT FIND NORMAL	Data in MASTER/ASTER Array inconsistent. Some routine has destroyed portions of MASTER/ASTER.

TABLE B.1

Body Type	3 Letter ID	11-20	21-30	31-40	41-50	51-60	61-70	No. of Cards Needed
Truncated Elliptic Cone	TEC	V_x	V_y	V_z	H_x	H_y	H_z	1 of 3
		N_x^*	N_y^*	N_z^*	A_x^*	A_y^*	A_z^*	2 of 3
		R1	R2	P				3 of 3
Torus	TOR	V_x	V_y	V_z	N_x	N_y	N_z	1 of 2
		R1	R2					2 of 2
Arbitrary Curved Surface	ARS	M	N					1 of n
		$X(1,1)$	$Y(1,1)$	$Z(1,1)$	$X(1,2)$	$Y(1,2)$	$Z(1,2)$	3 of n
		$X(1,N)$	$Y(1,N)$	$Z(1,N)$				N+2 of n
		$X(2,1)$						
		\vdots						
		$X(M,1)$						
		\vdots						
		$X(M,N)$	$Y(M,N)$	$Z(M,N)$				

Next page is blank.

APPENDIX C
SAM-C BENCHMARK PROBLEM INPUT

Next page is blank.

C.1 SHIELDING BENCHMARK PROBLEM 4.0 (Abridged)

Gamma-Ray Dose Above a
Plane Source of ^{60}Co on an
Air/Ground Interface

Original Submitted by:

Charles W. Garrett
Radiation Research Associates, Inc.*
8404 Westmont Court
Bethesda, Maryland 20034
2 December 1968

Accepted by BPG:

December, 1968

This Abridged Version prepared September, 1969

* Corporate offices located in Fort Worth, Texas.

Next page is blank.

"1. INTRODUCTION

The dose rate three feet above an air/ground interface contaminated with gamma-ray emitting isotopes is often used as a basic normalizing parameter in fallout radiation environment studies and fallout shielding methodology. For example, the dose rate three feet above a fallout field is the basic quantity to which geometric and barrier factors are applied in the currently-used "Engineering Method" (References 1,2*). This technique predicts shield effectiveness in fallout situations. There have been experimental measurements of the dose above fallout fields in several weapons test (References 3,4) and several calculations (References 1,5) of the same quantity have been made.

"However, because of the obvious difficulties associated with measurements of actual fallout, many studies have concerned themselves with the radiation environment above interface planes contaminated with a single isotope. In particular, ^{137}Cs and ^{60}Co have been extensively used in these investigations.

"2. THE AIR/GROUND INTERFACE PROBLEM

This benchmark problem is concerned with the theoretical computation of various quantities above an ideal air/ground interface uniformly contaminated with ^{60}Co . A discussion of experimental results is also included for comparative purposes.

"2.1 Problem Geometry

Figure C1 illustrates the geometry for the theoretical benchmark. A receiver (detector) point is located three feet above the air/ground interface which is assumed to be smooth and infinite in extent. ^{60}Co is uniformly distributed on the interface. Although ^{60}Co emits one 1.17 MeV photon and one 1.33 MeV photon per disintegration, many studies assume an average photon energy of 1.25 MeV. This assumption introduces negligible errors, and the benchmark data are normalized to a source strength of one 1.25 MeV photon emitted isotropically per cm^2 of interface area per second. [For SAM C the photon energy of 1.33 MeV was used.]

*References are not included in this abridged version. Ed.

BASED ON ORNL-DWG 69-2121 IN SHIELDING BENCHMARK PROBLEM 4.0.

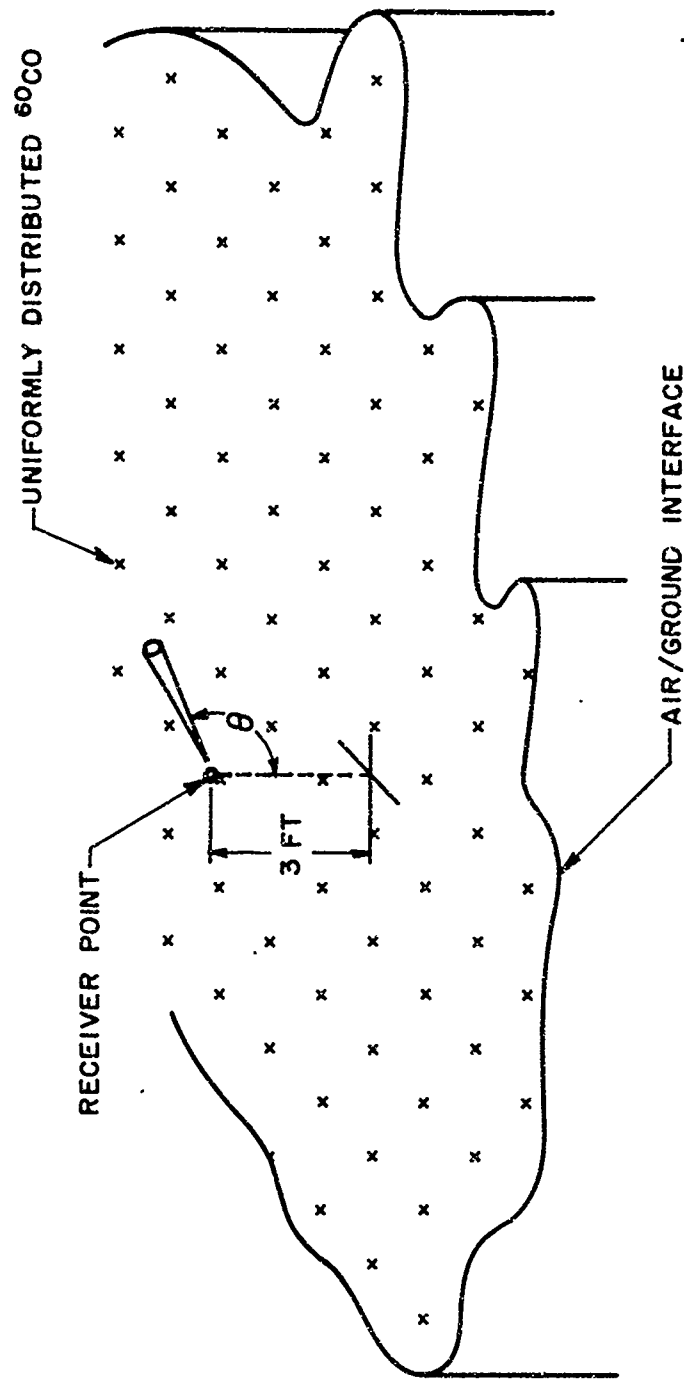


Figure C1. Air Over Ground Geometry

"Table I lists the constituents of the air and ground. The air density is $1.29 \times 10^{-3} \text{ g/cm}^3$; the ground is assumed to have a composition similar to Nevada Test Site soil (Reference 6).

TABLE I AIR AND GROUND COMPOSITIONS

	Element	Atomic Concentration (atoms/cm ³)
Air [*]	Nitrogen	4.19×10^{19}
	Oxygen	1.13×10^{19}
	Argon	2.53×10^{17}
Ground	Hydrogen	8.55×10^{21}
	Oxygen	2.27×10^{22}
	Aluminum	2.01×10^{21}
	Silicon	9.53×10^{21}

*There has been a slight change in format. Ed.

"2.2 Quantities Calculated

Quantities calculated at the receiver point are: (1) the total kerma rate,** T, in air; (2) the kerma rate,** D, in air from uncollided photons; (3) the dose buildup factor, B; and (4) the differential angle and energy distribution of the number flux density, $\phi(E, \theta)$, where θ is the receiver polar angle (Figure C1). The number flux density energy spectrum, I(E), [is the only quantity calculated by SAM C]"[calculated results appear in Tabular Form in Table IV and in graphical form in Figure C2].

** For the photon energies and geometry of this problem, the numerical difference between kerma rate and absorbed dose rate in air is small and can be ignored. Other studies quote kerma rate in tissue, and adjustments should be made to compare the results of such studies with this benchmark.

TABLE IV* SCATTERED PHOTON KERMA RATES THREE FEET ABOVE AN INFINITE
⁶⁰Co CONTAMINATED PLANE

Energy Interval (MeV)	K(E) (ergs=cm ² /g)	Flux Density (Photons/cm ² -sec)	Kerma Rate (ergs/g-sec)
.02 - .03	1.06(-8)	1.47(-3)	1.56(-11)
.03 - .04	5.28(-9)	1.55(-2)	8.18(-11)
.04 - .06	3.06(-9)	9.32(-2)	2.85(-10)
.06 - .10	3.06(-9)	2.10(-1)	6.43(-10)
.10 - .18	5.56(-9)	2.99(-1)	1.66(- 9)
.18 - .30	1.08(-8)	2.94(-1)	3.18(- 9)
.30 - .50	1.89(-8)	1.93(-1)	3.65(- 9)
.50 - .75	2.92(-8)	1.08(-1)	3.15(- 9)
.75 - 1.00	3.97(-8)	8.31(-2)	3.30(- 9)
1.00 - 1.25	5.00(-8)	1.69(-1)	8.45(- 9)
TOTALS:		1.47(0)	2.44(- 8)

NOTE: Read 1.06(-8) as 1.06×10^{-8}

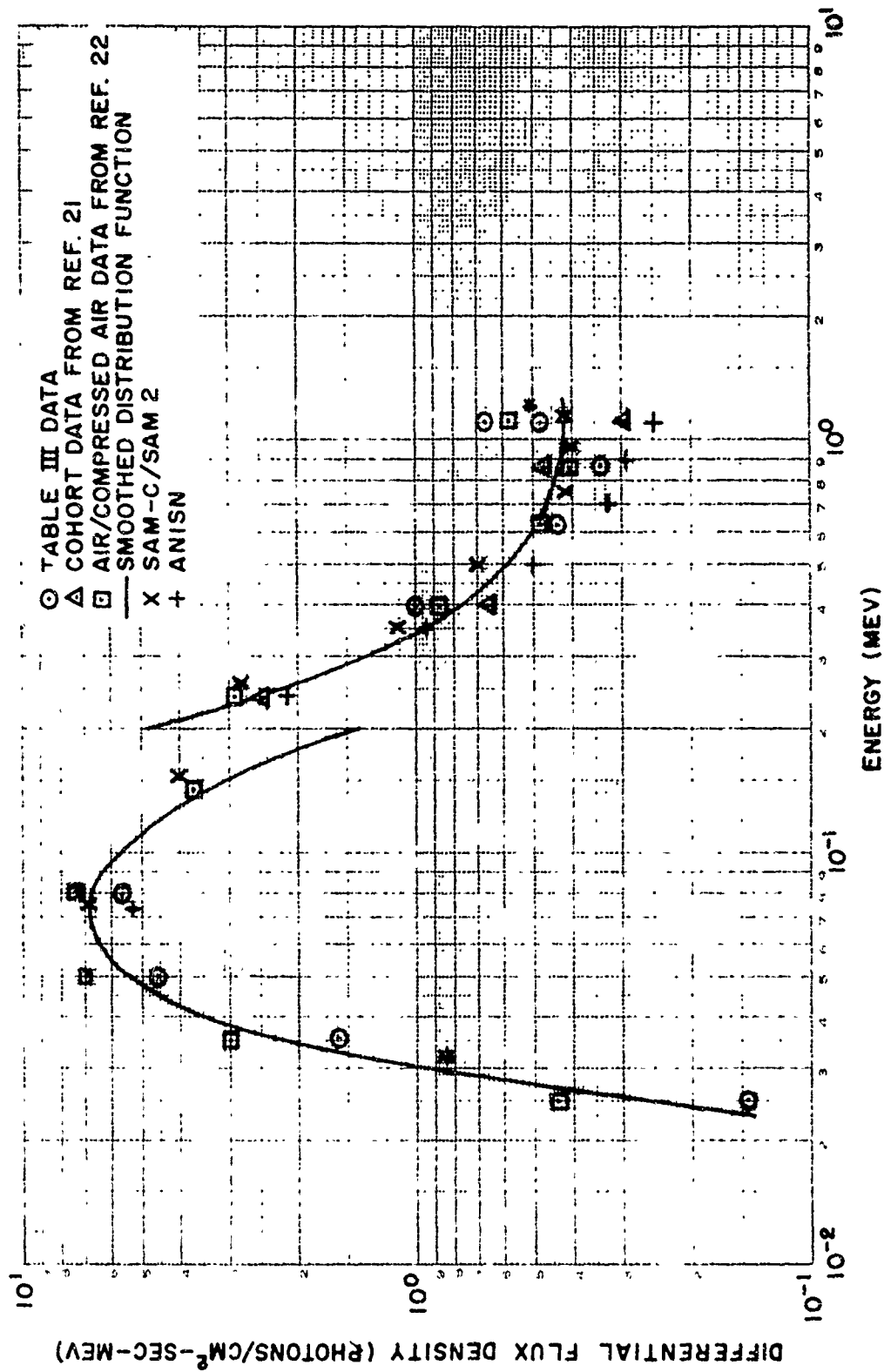
* Tables II and III have been omitted in this version. Ed.

[Results]

"On Figure C2 (Figure 6 in original document), differential scattered photon flux density energy spectra are plotted for the two cases shown on Figure 4,* along with data from a 7,000 history air/ground COHORT Monte Carlo study by French (Reference 21). Although neither the Table III or the COHORT data was smoothed in any way prior to constructing Figure 5, adjustments had to be made in the three lowest energy groups of the air/compressed air case. In that case, a severe fluctuation (visible on Figure 4) occurred in the 30° - 40° angle interval in each of the three energy bins. The solid curve on Figure 5 was obtained by intuitively smoothing all available data and, in addition, making use of two known facts; the magnitude of the discontinuity at the first scattering cutoff energy (0.212 MeV) and the value at the source energy (1.25 MeV). As described in Reference 24, these values can be easily and accurately computed. The value computed for the energy spectrum at 1.25 MeV is 0.43 photons/cm²-sec-MeV, and the magnitude of the discontinuity is 3.40 photons/cm²-sec-MeV.

* Figures 2, 3, 4, and 5 of original document have been omitted. Ed.

BASED ON ORNL-DWG 69-3429 IN SHIELDING BENCHMARK PROBLEM 4.0



* Plotted Incorrectly on Original.

Figure C2 Monte Carlo Scattered Flux Density Energy Spectra.

"In conclusion, it is to be emphasized that the differential data and smoothed curves presented on Figures 3, 4, and 5 contain rather large uncertainties, and must not be taken as absolute standards...."

C.2 AIR FORCE WEAPON LAB "ROCKET" GEOMETRY

Next page is blank.

FURTRAN COILING FORM AND DATA SHEET

SAMC GEOMETRY INPUT

A KRIS WIDDISON

DATE: 10/10/80

TEST GEOMETRY -- INCLUDES ALL TYPES OF SOLIDS									
	I	II	III	IV	V	VI	VII	VIII	IX
-250 0	150 0	-50 0	450 0						
-50 0	250 0	450 0							
2.0									
* RAW	0 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0
	0 0	60 0	0 0	0 0	100 0	0 0	0 0	0 0	0 0
* BOX	0 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0
	0 0	60 0	0 0	0 0	-80 0	0 0	0 0	0 0	0 0
* REC	-100 0	30 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0
	0 0	15 0	0 0	0 0	10 0	0 0	0 0	0 0	0 0
* RCIC	-30 0	40 0	20 0	0 0	0 0	0 0	0 0	0 0	150 0
	10 0								
* TRC	-30 0	40 0	170 0	0 0	0 0	0 0	0 0	0 0	10 0
	10 0	0 0							
* ARB	-20 0	0 0	20 0	-20 0	20 0	20 0	20 0	20 0	20 0
	-20 0	0 0	220 0	-20 0	220 0	20 0	20 0	20 0	220 0
	-60 0	0 0	20 0	-60 0	20 0	20 0	20 0	20 0	220 0
	-60 0	0 0	220 0	-60 0	220 0	20 0	20 0	20 0	220 0
	1234 2684 6578 1375 3487 1562								
* SPH	-30 0	40 0	172 0	1 0					
* EL	-30 0	40 0	140 0	-30 0	30 0	150 0			
	20 0								
* SPH	-30 0	40 0	30 0	5 0					

APPENDIX D
LISTING OF THE AMSAA OCTOBER REVISED STANDARD MAGIC PROGRAM

Next page is blank.

C	DIMENSION MASTER(30000),A(6)	MAIN	1
	COMMON ASTER(30000)	MAIN	2
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERK,DIST	MAIN	3
	COMMON/UNCSEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	MAIN	4
	1 LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	MAIN	5
	COMMON/TEMPOR/XS(6),X(6),IX(8),IT(10),IA(9),IN(9)	MAIN	6
	COMMON/WALT/LIRFC,NG1ERR	MAIN	7
	COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPE8,NO,IYES	MAIN	8
	COMMON/ENGEOM/LFGEOM	MAIN	9
	COMMON/SIZE/NDQ	MAIN	10
	COMMON/ERR/IERR	MAIN	11
	COMMON/XANDM/IRANDM	MAIN	12
	EQUIVALENCE (ASTLR,MASTER)	MAIN	13
C	901 FORMAT(1H1,32HTHIS IS THE 11 APR 69 VERSION OF /	MAIN	14
	1 1H,32HTHE BRLESC MAGIC PROGRAM ***** //)	MAIN	15
	902 FORMAT(16H BEGIN EXECUTION)	MAIN	16
	903 FORMAT(P110)	MAIN	17
	904 FORMAT(1H0,10X,42HTHE TAPE 4 USED FOR THIS RUN HAS THE TITLE /	MAIN	18
	1 10A6/)	MAIN	19
	905 FORMAT(1H0,10HENTER GENI)	MAIN	20
	906 FORMAT(1H0,12HLEAVING GENI)	MAIN	21
	907 FORMAT(1H0,35HTERMINATION ON GEOMETRY INPUT ERROR,5X,5HIERR=,15)	MAIN	22
	908 FORMAT(1H1,15HTESTG IS CALLED)	MAIN	23
	909 FORMAT(1H0,13HLEAVING TESTG)	MAIN	24
	910 FORMAT(1H1,24HREGION TYPE DATA FOLLOWS, 8X,6H1IRFO=,110/	MAIN	25
	1 1H,6HREGION,6X,4HCODE,6X,4HIYPE,6X,11HDESCRIPTION/)	MAIN	26
	911 FORMAT(3110,10X,6A6)	MAIN	27
	912 FORMAT(16,110,19,7X,6A6)	MAIN	28
	913 FORMAT(1H0,23HNO ROOM FOR IDENT TABLE,5X,7HLEGEOM=,17,5X,	MAIN	29
	1 6H1IRFO=,17)	MAIN	30
	914 FORMAT(1H0,32HWRITE TAPE 1 OPTION IS SPECIFIED)	MAIN	31
	915 FORMAT(15,10X,10A6)	MAIN	32
	916 FORMAT(1H1,11HENTER VOLUM)	MAIN	33
	917 FORMAT(1H0,13HLEAVING VOLUM)	MAIN	34
	918 FORMAT(1H,6H 999.9)	MAIN	35
	919 FORMAT(1H1,11HEND OF CASE,15)	MAIN	36
	925 FORMAT(1H1,32HNUM OF ASPECT ANGLES FOR GRID IS,15)	MAIN	37
	927 FORMAT(1015)	MAIN	38
	928 FORMAT(1H1,32HNUM OF ASPECT ANGLES FOR AREA IS,15)	MAIN	39
	929 FORMAT(1H0,31HNUMBER OF G1 ERRORS ENCOUNTERED,15)	MAIN	40
	930 FORMAT(1H0,31HNUMBER OF 0 ITEMS ENCOUNTERED,15)	MAIN	41
	999 FORMAT(1H0,10HEND OF RUN)	MAIN	42
C	IRANDM=0	MAIN	43
	WRITE (5,901)	MAIN	44
	WRITE (6,902)	MAIN	45
C	I15=2**15	MAIN	46
	I30=2**30	MAIN	47
	PINF=1.0E50	MAIN	48
	NU=0	MAIN	49
	IYES=1	MAIN	50
	IERK=C	MAIN	51
	LBASE=1	MAIN	52
	KLOOP=0	MAIN	53
	NDQ=30000	MAIN	54
C	READ (5,903)IRDTP4,IWRIP4,ITESTG,IRAYSK,ICARDI,IENTLV,IVOLUM	MAIN	55
		MAIN	56
		MAIN	57
		MAIN	58
		MAIN	59
		MAIN	60

IF(IKOTP4.NE.0)IKOTP4=IYES	MAIN 61
IF(IWOTP4.NE.0)IWOTP4=IYES	MAIN 62
IF(ITESTG.NE.0)ITESTG=IYES	MAIN 63
IF(IRAYSK.NE.0)IRAYSK=IYES	MAIN 64
IF(ICARDI.NE.0)ICARDI=IYES	MAIN 65
IF(IENTLV.NE.0)IENTLV=IYES	MAIN 66
IF(IVOLUM.NE.0)IVOLUM=IYES	MAIN 67
C	MAIN 68
IF(IKOTP4.EQ.NO) GOTO 10	MAIN 69
READ (4) DUMMY,ASTER,LBASE,PINF,IERR,NRPP,NTRIP,NSCAL,	MAIN 70
1 NBODY,NRMAX,LTRIP,LSCAL,LREGD,LDATA,LRIN,LROT,LIO,LOCDA,	MAIN 71
2 LBODY,LIRFO,SCALE,LRI,LRO,PINF,IT	MAIN 72
WRITE (6,904)(IT(I),I=1,10)	MAIN 73
GOTO 20	MAIN 74
C	MAIN 75
C PROCESS GEOMETRY	MAIN 76
C	MAIN 77
10 DO 11 I=LBASE,NDQ	MAIN 78
ASTER(I)=0.	MAIN 79
11 CONTINUE	MAIN 80
C	MAIN 81
WRITE (6,905)	MAIN 82
CALL GENI	MAIN 83
WRITE (6,906)	MAIN 84
IERR=0	MAIN 85
IF(IERR.LE.0)GOTO 12	MAIN 86
WRITE (6,907)IERR	MAIN 87
STOP	MAIN 88
C	MAIN 89
12 IF(IWOTP4.EQ.NO)GOTO 20	MAIN 90
WRITE(4) DUMMY,ASTER,LBASE,PINF,IERR,NRPP,NTRIP,NSCAL,	MAIN 91
1 NBODY,NRMAX,LTRIP,LSCAL,LREGD,LDATA,LRIN,LROT,LIO,LOCDA,	MAIN 92
2 LBODY,LIRFO,SCALE,LRI,LRO,PINF,IT	MAIN 93
C	MAIN 94
C TEST G	MAIN 95
C	MAIN 96
20 IF(ITESTG.EQ.NO)GOTO 30	MAIN 97
WRITE (6,908)	MAIN 98
CALL TESTG	MAIN 99
WRITE (6,909)	MAIN 100
ITESTG=NO	MAIN 101
C	MAIN 102
C VOLUM	MAIN 103
C	MAIN 104
30 IF(IVOLUM.EQ.NO)GOTO 40	MAIN 105
WRITE (6,916)	MAIN 106
CALL VOLUM	MAIN 107
WRITE (6,917)	MAIN 108
IVOLUM=NO	MAIN 109
C	MAIN 110
C REGION TYPE DATA / ICODE / IDENT /	MAIN 111
C	MAIN 112
C IRN = REGION NUMBER	MAIN 113
C ICODE = ITEM CODE	MAIN 114
C IDENT = SPACE CODE AND SPECIAL IDENTIFICATION	MAIN 115
C 0 NO IDENT CODE	MAIN 116
C 10,20,30,40,50,60,70,80,90 SPECIAL IDENTIFICATION	MAIN 117
C SKIRT=10 ARMOR=20 TARGET=30	MAIN 118
C -1,1-9,11-19,21-29,.....,91-99 SPACE CODES	MAIN 119
C	MAIN 120

40	LIRFO=NDQ-NRMAX-10	MAIN 121
	IF(LIRFO.GT.LEGEOM)GOTO 41	MAIN 122
	WRITE(6,913)LEGEOM,LIRFO	MAIN 123
	STOP	MAIN 124
41	WRITE (6,910)LIRFO	MAIN 125
	MASTER(LIRFO-1)=115+1	MAIN 126
C		MAIN 127
42	READ(5,911)IRN,ICODE,IDENT,(A(I),I=1,6)	MAIN 128
	IF(IRN.LE.0)GOTO 50	MAIN 129
	WRITE (6,912)IRN,ICODE,IDENT,(A(I),I=1,6)	MAIN 130
	IDENT=IDENT+1	MAIN 131
	K=LIRFO+IRN-1	MAIN 132
	MASTER(K)=ICODE*115+IDENT	MAIN 133
	GOTO 42	MAIN 134
C		MAIN 135
C	NOAA = NUM OF ASPECT ANGLES FOR GRID	MAIN 136
C	ITAPE8 IS THE SUPPRESS PRINTER OPTION	MAIN 137
C	IWO1 IS WRITE OPTION FOR TAPE 1	MAIN 138
C	NAREA = NUM OF ASPECT ANGLES FOR AREA	MAIN 139
C		MAIN 140
50	READ (5,927)NOAA,IWOT,ITAPE8,NAREA	MAIN 141
	IF(IWOT.NE.0)IWOT=1YES	MAIN 142
	IF(ITAPE8.EQ.0)GOTO 51	MAIN 143
	ITAPE8=NO	MAIN 144
	GOTO 52	MAIN 145
51	ITAPE8=1YES	MAIN 146
52	IF(IWOT.EQ.NO)GOTO 60	MAIN 147
	REWIND 1	MAIN 148
	WRITE (6,914)	MAIN 149
	WRITE(1,915)NOAA,(IT(I),I=1,10)	MAIN 150
C		MAIN 151
C	GRID	MAIN 152
C		MAIN 153
60	IF(NOAA.LE.0)GOTO 70	MAIN 154
	WRITE (6,925)NOAA	MAIN 155
C		MAIN 156
	DO 61 I=1,NOAA	MAIN 157
	CALL GRID	MAIN 158
	IF(IWOT.EQ.1YES)WRITE(1,918)	MAIN 159
	WRITE (6,919)I	MAIN 160
	WRITE (6,929)IERR	MAIN 161
	WRITE (6,930)IERR0	MAIN 162
	IERR=0	MAIN 163
	IERR0=0	MAIN 164
61	CONTINUE	MAIN 165
C		MAIN 166
C	AREA	MAIN 167
C		MAIN 168
70	IF(NAREA.LE.0)GOTO 99	MAIN 169
	WRITE (6,928)NAREA	MAIN 170
C		MAIN 171
	DO 71 I=1,NAREA	MAIN 172
	CALL AREA	MAIN 173
	WRITE (6,919)I	MAIN 174
	IERR=0	MAIN 175
71	CONTINUE	MAIN 176
C		MAIN 177
99	WRITE (6,999)	MAIN 178
	STOP	MAIN 179
	END	MAIN 180

C		MAIN 181
C		MAIN 182
	SUBROUTINE UN2(L,J1,J2)	**** 1
C	UNPACKS 2 ITEMS FROM THE MASTER-ASTER ARRAY	UN2 2
	COMMON MASTER(30000)	UN2 3
	I3=MASTER(L)	UN2 4
	J1=I3/32768	UN2 5
	J2=I3-J1*32768	UN2 6
	RETURN	UN2 7
	END	UN2 8
C		UN2 9
C		UN2 10
	SUBROUTINE UN3(L,J1,J2,J3)	**** 2
C	UNPACKS 3 ITEMS FROM G1 STORAGE	UN3 2
	COMMON MASTER(30000)	UN3 3
	I3=MASTER(L)	UN3 4
	I2=I3/32768	UN3 5
	J1=I2/64	UN3 6
	J2=I2-J1*64	UN3 7
	J3=I3-I2*32768	UN3 8
	RETURN	UN3 9
	END	UN3 10
C		UN3 11
C		UN3 12
	SUBROUTINE OPENK(L,J1,J2,J3)	**** 3
	COMMON/UTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IKSTRT,IENC,	OPENK 2
1	ITR(200),CA,CE,SA,SE	OPENK 3
C		OPENK 4
C	UNPACKS 3 ITEMS FROM COMPONENT LINE OF SIGHT STORAGE ITR	OPENK 5
C	/ SURFACE NUM / BODY NUM / NEXT REGION /	OPENK 6
C		OPENK 7
	I3=ITR(L)	OPENK 8
	I2=I3/4096	OPENK 9
	J1=I2/4096	OPENK 10
	J2=I2-J1*4096	OPENK 11
	J3=I3-I2*4096	OPENK 12
	RETURN	OPENK 13
	END	OPENK 14
C		OPENK 15
C		OPENK 16
	FUNCTION RAN(M)	**** 4
	COMMON/RANDM/IRN	RAN 2
C	GENERATES RANDOM NUMBERS	RAN 3
	RAN=URAN31(IRN)	RAN 4
	RETURN	RAN 5
	END	RAN 6
C		RAN 7
C		RAN 8
	FUNCTION URAN31(I)	**** 5
	IF(I)20,10,20	URAN31 2
10	I=11111111	URAN31 3
20	J=I	URAN31 4
	J=J*25	URAN31 5
	J=J-(J/67108864)*67108864	URAN31 6
	J=J*25	URAN31 7
	J=J-(J/67108864)*67108864	URAN31 8
	J=J*5	URAN31 9
	J=J-(J/67108864)*67108864	URAN3110
	A1=J	URAN3111
	I=J	URAN3112

URAN31=A1/67108964.	URAN3113
RETURN	URAN3114
END	URAN3115
C	URAN3116
C	URAN3117
SUBROUTINE CROSS(ANSWER,FIRST,SECOND)	**** 6
DIMENSION ANSWER(3),FIRST(3),SECOND(3)	CROSS 2
C	CROSS 3
COMPUTES CROSS PRODUCT ANSWER = FIRST X SECOND	CROSS 4
ANSWER(1) = FIRST(2)*SECOND(3) - FIRST(3)*SECOND(2)	CROSS 5
ANSWER(2) = FIRST(3)*SECOND(1) - FIRST(1)*SECOND(3)	CROSS 6
ANSWER(3) = FIRST(1)*SECOND(2) - FIRST(2)*SECOND(1)	CROSS 7
RETURN	CROSS 8
END	CROSS 9
C	CROSS 10
C	**** 7
FUNCTION DOT(FIRST,SECOND)	DOT 2
DIMENSION FIRST(3),SECOND(3)	DOT 3
C	DOT 4
COMPUTES DOT PRODUCT DOT = FIRST . SECOND	DOT 5
DOT = FIRST(1)*SECOND(1)+FIRST(2)*SECOND(2)+FIRST(3)*SECOND(3)	DOT 6
RETURN	DOT 7
END	DOT 8
C	**** 8
C	UNIT 2
SUBROUTINE UNIT(V)	UNIT 3
DIMENSION V(3)	UNIT 4
C	UNIT 5
COMPUTES UNIT VECTOR	UNIT 6
TEMP = SQRT(DOT(V,V))	UNIT 7
V(1)=V(1)/TEMP	UNIT 8
V(2)=V(2)/TEMP	UNIT 9
V(3)=V(3)/TEMP	UNIT 10
RETURN	UNIT 11
END	**** 9
C	QRTIC 2
C	QRTIC 3
SUBROUTINE QRTIC(C,R,NRE)	QRTIC 4
C	QRTIC 5
SOLVES A POLYNOMIAL EQUATION OF THE TYPE	QRTIC 6
C	QRTIC 7
$X^4 + C(1)X^3 + C(2)X^2 + C(3)X + C(4) = 0$	QRTIC 8
C	QRTIC 9
THE COEFFICIENT OF X^4 IS ASSUMED TO BE 1	QRTIC 10
C	QRTIC 11
R CONTAINS THE ROOTS	QRTIC 12
C	QRTIC 13
NRE CONTAINS THE NUMBER OF REAL ROOTS	QRTIC 14
C	QRTIC 15
IF THERE ARE TWO REAL ROOTS THEY WILL BE IN R(1) AND R(2),	QRTIC 16
C	QRTIC 17
WITH THE COMPLEX ROOTS R(3) +/- R(4)*I	QRTIC 18
C	QRTIC 19
IF THERE ARE NO REAL ROOTS, THE COMPLEX ROOTS ARE	QRTIC 20
C	QRTIC 21
R(1) +/- R(2)*I AND R(3) +/- R(4)*I	QRTIC 22
C	QRTIC 23
DIMENSION C(4),R(4),CP(3),Y(3)	QRTIC 24
C1SQ=C(1)**2	QRTIC 25
CP(1)=-C(2)	QRTIC 26
CP(2)=C(1)*C(3)-4.*C(4)	
CP(3)=(4.*C(2)-C1SQ)*C(4)-C(3)**2	
CALL CUBIC(CP,Y,NRE)	
A=C1SQ/4.-C(2)+Y(1)	
B=.5*C(1)*Y(1)-C(3)	
D=.25*Y(1)**2-C(4)	
IF(A.GT.0.)GOTO 10	
E=0.	
GOTO 20	
10 E=SQRT(A)	
20 IF(D.GT.0.)GOTO 30	

F=0.	QRTIC 27
GOTO 50	QRTIC 28
30 F=SIGN(SQRT(D),B)	QRTIC 29
50 NKF=0	QRTIC 30
REAL=-.25*C(1)+.5*L	QRTIC 31
DSCR=REAL**2-.5*Y(1)+F	QRTIC 32
RAD=SQRT(ABS(DSCR))	QRTIC 33
IF(DSCR.LT.0.)GOTO 60	QRTIC 34
NRE=2	QRTIC 35
R(1)=REAL+RAD	QRTIC 36
R(2)=REAL-RAD	QRTIC 37
GOTO 65	QRTIC 38
60 R(3)=REAL	QRTIC 39
R(4)=RAD	QRTIC 40
65 REAL=REAL-F	QRTIC 41
DSCR=REAL**2-.5*Y(1)-F	QRTIC 42
RAD=SQRT(ABS(DSCR))	QRTIC 43
IF(DSCR.LT.0.)GOTO 80	QRTIC 44
NRE=NRE+2	QRTIC 45
R(NRE)=REAL-RAD	QRTIC 46
R(NRE-1)=REAL+RAD	QRTIC 47
RETURN	QRTIC 48
80 R(NRE+1)=REAL	QRTIC 49
R(NRE+2)=RAD	QRTIC 50
RETURN	QRTIC 51
END	QRTIC 52
C	QRTIC 53
C	QRTIC 54
SUBROUTINE CUBIC(C,R,NRE)	**** 10
C	CUBIC 2
C	CUBIC 3
C	CUBIC 4
C	CUBIC 5
C	CUBIC 6
C	CUBIC 7
C	CUBIC 8
C	CUBIC 9
C	CUBIC 10
C	CUBIC 11
C	CUBIC 12
C	CUBIC 13
C	CUBIC 14
C	CUBIC 15
C	CUBIC 16
C	CUBIC 17
C	CUBIC 18
C	CUBIC 19
C	CUBIC 20
C	CUBIC 21
C	CUBIC 22
C	CUBIC 23
C	CUBIC 24
C	CUBIC 25
C	CUBIC 26
C	CUBIC 27
C	CUBIC 28
C	CUBIC 29
C	CUBIC 30
C	CUBIC 31
C	CUBIC 32
DIMENSION C(3),R(3)	
C1SQ=C(1)**2	
P=C(2)-C1SQ/3.	
Q=C(3)-C(1)*(C(2)/3.-2.*C1SQ/27.)	
DEL=4.*P**3+27.*Q**2.	
T=C(1)/3.	
IF(DEL.LT.0.)GOTO 10	
SQ=SQRT(DEL/108.)	
HQ=.5*Q	
A=-HQ+SQ	
B=-HQ-SQ	
CRTA=SIGN(ABS(A)**.3333333333333333,A)	
CRTB=SIGN(ABS(B)**.3333333333333333,B)	
Y=CRTA+CRTB	
R(1)=Y-T	
R(2)=-.5*Y-T	
R(3)=.866025404*(CRTA-CRTB)	
NRE=1	
RETURN	
10 PHI3=ATAN2(SQRT(-DEL/27.),-Q)/3.	
CON=2.*SQRT(-P/3.)	
R(1)=CON*COS(PHI3)-T	

	R(2)=-CON*COS(1.04719755-PHI3)-T	CUBIC 33
	R(3)=-CON*COS(1.04719755+PHI3)-T	CUBIC 34
	NRE=3	CUBIC 35
	RETURN	CUBIC 36
	END	CUBIC 37
C		CUBIC 38
C		CUBIC 39
	FUNCTION XDIST(XA,XB)	**** 11
C	COMPUTES THE DISTANCE BETWEEN XA AND XB	XDIST 2
	DIMENSION XA(3),XB(3)	XDIST 3
	XSUM=0.	XDIST 4
	DO 10 I=1,3	XDIST 5
	XSUM=XSUM+(XA(I)-XB(I))**2	XDIST 6
10	CONTINUE	XDIST 7
	XDIST=SQRT(XSUM)	XDIST 8
	RETURN	XDIST 9
	END	XDIST 10
C		XDIST 11
C		XDIST 12
	SUBROUTINE DCOSP(XA,XB,WA)	**** 12
C	COMPUTES DIRECTION COSINES FROM POINT XA TO POINT XB	DCOSP 2
C	AND STORES DIRECTION COSINES IN WA	DCOSP 3
	DIMENSION XA(3),XB(3),WA(3)	DCOSP 4
	DIS=XDIST(XA,XB)	DCOSP 5
	DO 10 I=1,3	DCOSP 6
	WA(I)=(XB(I)-XA(I))/DIS	DCOSP 7
10	CONTINUE	DCOSP 8
	RETURN	DCOSP 9
	END	DCOSP 10
C		DCOSP 11
C		DCOSP 12
	SUBROUTINE TROPIC(WP)	**** 13
C	GENERATES RANDOM DIRECTION COSINES FROM AN	TROPIC 2
C	ISOTROPIC DISTRIBUTION	TROPIC 3
	DIMENSION WP(3)	TROPIC 4
10	X1=РАН (-1)	TROPIC 5
	X2=РАН (-1)	TROPIC 6
	X1S=X1**2	TROPIC 7
	X2S=X2**2	TROPIC 8
	T=X1S+X2S	TROPIC 9
	IF(T.GE.1.)GOTO 10	TROPIC10
C	CALC SIN AND COS OF A RANDOM ANGLE PHI	TROPIC11
	CSPHI=(X1S-X2S)/T	TROPIC12
	SNPHI=(2.*X1*X2)/T	TROPIC13
	X1=РАН (-1)	TROPIC14
	IF(X1.LE..5)SNPHI=-SNPHI	TROPIC15
C	CALC COS AND SIN OF RANDOM ANGLE TH	TROPIC16
	CSTHT=2.*РАН (-1)-1.	TROPIC17
	SNHTT=SQRT(1.-CSTHT**2)	TROPIC18
C	CALC DIRECTION COSINES	TROPIC19
	WP(1)=SNHTT*SNPHI	TROPIC20
	WP(2)=SNHTT*CSPHI	TROPIC21
	WP(3)=CSTHT	TROPIC22
	RETURN	TROPIC23
	END	TROPIC24
C		TROPIC25
C		TROPIC26
C		TROPIC27
C		TROPIC28
	SUBROUTINE GFNI	**** 14

DIMENSION MASTER(30000),ITY(11),IAN(8),IAA(8),FX(20),	GENI	2
1 NOC(3),NO1(3),NO2(3),O4(3),TT(3),TT1(3),TT2(3),NBOD(11)	GENI	3
COMMON ASTER(30000)	GENI	4
COMMON/GFOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	GENI	5
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	GENI	6
1 LDATA,LRIN,LROT,LIO,LUCDA,I15,I30,LBODY,NASC,KLOOP	GENI	7
COMMON/TEMPOR/XS(6),X(6),IX(8),IT(10),IA(9),IN(9)	GENI	8
COMMON/CONTRL/ITESTG,IRAYSK,IENLV,IVOLUM,IWOT,ITAPE8,NO,IYES	GENI	9
COMMON/SIZE/NUQ	GENI	10
COMMON/UNCLE/MN,IC(4)	GENI	11
COMMON/RRPP/LRPPD,LABUT	GENI	12
EQUIVALLNCE(ASTER,MASTER)	GENI	13
901 FORMAT(1H0,24HSTART READING SOLID DATA)	GENI	14
902 FORMAT(10A6)	GENI	15
903 FORMAT(1H0,10A6/)	GENI	16
904 FORMAT(7I10)	GENI	17
905 FORMAT(4X,34HNO. OF RECTANGULAR PARALLELEPIPEDS,I10/	GENI	18
1 4X,34HNO. OF TRIPLETS ,I10/	GENI	19
2 4X,34HNO. OF SCALERS ,I10/	GENI	20
3 4X,34HNO. OF SOLIDS ,I10/	GENI	21
4 4X,34HMAX NO. OF REGIONS ,I10)	GENI	22
906 FORMAT(1H0,45X,32HRECTANGULAR PARALLELEPIPED INPUT)	GENI	23
907 FORMAT(1H0,37X,12HTRIPLET DATA)	GENI	24
908 FJRMAT(6E12.6)	GENI	25
909 FORMAT(18,17X,3F12.5)	GENI	26
910 FORMAT(1H0,25X,12HSCALAR DATA)	GENI	27
911 FORMAT(1H0,50X,22HDESCRIPTION OF SOLIDS)	GENI	28
912 FORMAT(3A1,A3,A4,6F10.5)	GENI	29
913 FORMAT(1H0,6HITYPE ,A3,27H DOES NOT MATCH WITH AN ITY)	GENI	30
914 FORMAT(19,1X,3A1,3X,A3,A4,3X,8I5)	GENI	31
915 FORMAT(18,1X,3A1,2X,A3,A4,4X,6F12.5)	GENI	32
916 FORMAT(25X,6F12.5)	GENI	33
917 FORMAT(1H0,38HNO MORE ROOM FOR SOLID DATA LDATA=,I10,	GENI	34
1 5X,5HLEBOT=,I10,5X,4HNDQ=,I10)	GENI	35
918 FORMAT(1H0,25HFINISH READING SOLID DATA)	GENI	36
919 FORMAT(1H0, 5HLEGD,7H LREGD,7H LENLV,7H LRIN,7H LROT,	GENI	37
1 7H LIO,7H LECEOM/I5,6I7)	GENI	38
920 FORMAT(1H1,36X,23HREGION COMBINATION DATA)	GENI	39
921 FORMAT(15,1X,9(A2,I5))	GENI	40
922 FORMAT(1H0,30HERROR IN DESCRIPTION OF REGION,I5,	GENI	41
19H IN FIELD,I2,5X,24HBODY NUM.GT.NRPP + NBODY)	GENI	42
923 FORMAT(10X,9(1H(,A2,I5,1H),1X))	GENI	43
924 FORMAT(18,2X,9(1H(,A2,I5,1H),1X))	GENI	44
925 FORMAT(1H0,30HILLEGAL OPERATOR IN ABOVE CARD,5X,A2,	GENI	45
1 9H IN FIELD,I2)	GENI	46
926 FORMAT(1H0,29HERROR IN REGION INPUT IR=,I5,14H OR N.GT.NRMAX)	GENI	47
927 FORMAT(1H0,39HNO MORE ROOM FOR REGION DATA LDATA=,I10,	GENI	48
1 5X,4HNDQ=,I10)	GENI	49
928 FORMAT(1H0,26HFINISH READING REGION DATA)	GENI	50
929 FORMAT(14H ERROR, REGION,I10,18H IS PART OF REGION,I10)	GENI	51
930 FORMAT(24H FINISH CHECKING REGION ,I5)	GENI	52
931 FORMAT(1H0,34HNO MORE ROOM FOR ENTER LEAVE TABLE,5X,	GENI	53
1 6HLEDATA=,I10,5X,4HNDQ=I10,5X,4HPASS,I2,5X,3HIR=,I10)	GENI	54
932 FORMAT(1H0,28HTOTAL ROOM FOR GEOMETRY DATA,5X,7HLECEOM=,I6)	GENI	55
933 FORMAT(1H0,5HENTER,18I6/(23X,15I6))	GENI	56
934 FORMAT(1H ,5HLEAVE,18I6/(23X,15I6))	GENI	57
935 FORMAT(1H1,50X,18HBEGIN ARRAY OUTPUT/)	GENI	58
936 FORMAT(3(3I6,1X,E11.4,3H \$))	GENI	59
937 FORMAT(/)	GENI	60
	GENI	61

938	FORMAT(1H0,34HFINISH A PASS OF ENTER LEAVE TABLE,15)	GENI	62
939	FORMAT(1H0,14HERROR IN INPUT,5X,A3,23H DOES NOT ALLOW TRIPLET,	GENI	63
1	22H AND SCALAR TYPE INPUT)	GENI	64
940	FJRMAT(10X,6F10.5)	GENI	65
941	FORMAT(1H0,37HTERMINATION ON BAD REGION DESCRIPTION)	GENI	66
942	FORMAT(1H0,32HERROR IN DESCRIPTION OF BODY NUM,16/	GENI	67
1	7H VECTOR,3F12.5,24H IS NOT PERPENDICULAR TO /	GENI	68
2	7H VECTOR,3F12.5/)	GENI	69
943	FORMAT(1H0,27HERROR IN DESCRIPTION OF TOR,5X,8HR2.GT.R1/)	GENI	70
944	FORMAT(1H0,27HERROR IN DESCRIPTION OF TRC,5X,7HR1 = R2/)	GENI	71
945	FORMAT(1H0,5HBASE,7H LRPPD,	GENI	72
1	7H LABUT,7H LBODY,7H LBOO,7H LDATA,7H LBOT,7H LSCAL,	GENI	73
2	7H LTRIP,7H NDQ/15,9I7)	GENI	74
946	FORMAT(1H1,17HENTER-LEAVE TABLE)	GENI	75
947	FORMAT(1H0,11(2X,A3)/1115)	GENI	76
948	FORMAT(1H0,27HERROR IN DESCRIPTION OF TEC,5X,	GENI	77
1	41HHEIGHT VECTOR IS PARALLEL TO BASE ELLIPSE)	GENI	78
C		GENI	79
	INTEGER HHBOX,HHSPH,HHRCC,HHREC,HHTRC,HHELL,HHRAW,HHARB,HHTEC,	GENI	80
1	HHTOR,HHARS,HHOR,HHBR,HHR,HHRA,HHAR,HHBA,HHA,HHB	GENI	81
C		GENI	82
	DATA HHBOX,HHSPH,HHRCC,HHREC,HHTRC,HHELL,HHRAW,HHARB,	GENI	83
1	HHTEC,HHTOR,HHARS,HHBOX,3HSPH,3HRC,3HREC,3HTRC,	GENI	84
23	HELL,3HRAW,3HARB,3HTEC,3HTOR,3HARS/	GENI	85
DATA	HHOR,HHBR,HHR,HHRA,HHAR,HHBA,HHA,HHB	GENI	86
1/2	HOR,2HRR,1HR,2HRA,2HAR,2H A,2HA,2H /	GENI	87
	I1Y(1)=HHBOX	GENI	88
	I1Y(2)=HHSPH	GENI	89
	I1Y(3)=HHRCC	GENI	90
	I1Y(4)=HHREC	GENI	91
	I1Y(5)=HHTRC	GENI	92
	I1Y(6)=HHELL	GENI	93
	I1Y(7)=HHRAW	GENI	94
	I1Y(8)=HHARB	GENI	95
	I1Y(9)=HHTEC	GENI	96
	I1Y(10)=HHTOR	GENI	97
	I1Y(11)=HHARS	GENI	98
	I1A(1)=1	GENI	99
	I1A(2)=1	GENI	100
	I1A(3)=1	GENI	101
	I1A(4)=2	GENI	102
	I1A(5)=2	GENI	103
	I1A(6)=3	GENI	104
	I1A(7)=3	GENI	105
	I1A(8)=4	GENI	106
	I1A(1)=HHOR	GENI	107
	I1A(2)=HHBR	GENI	108
	I1A(3)=HHR	GENI	109
	I1A(4)=HHRA	GENI	110
	I1A(5)=HHAR	GENI	111
	I1A(6)=HHBA	GENI	112
	I1A(7)=HHA	GENI	113
	I1A(8)=HHB	GENI	114
	I1L=HHB	GENI	115
		GENI	116
C		GENI	117
	WRITE (6,901)	GENI	118
	READ(5,902)(I1(I),I=1,10)	GENI	119
	WRITE (6,903)(I1(I),I=1,10)	GENI	120
		GENI	121

READ(5,904)NRPP,NTRIP,NSCAL,NBODY,NRMAX,IPRIN,IRCHK	GENI 122
WRITE (6,905)NRPP,NTRIP,NSCAL,NBODY,NRMAX	GENI 123
C	GENI 124
C RPP	GENI 125
C	GENI 126
WRITE (6,906)	GENI 127
LAR=1	GENI 128
IF(NRPP.LE.0)GOTO 20	GENI 129
CALL RPPIN(LAR)	GENI 130
IF(I-RR.GT.0)RETURN	GENI 131
C	GENI 132
C LBODY STORAGE RESERVE 3*(NRPP+NBODY) WORDS	GENI 133
C / ITYPE / LDATA /	GENI 134
C / LOC ENTER LIST / LOC LEAVE LIST /	GENI 135
C / NUM ENTER / NUM LEAVE /	GENI 136
C	GENI 137
C LDATA POINTS TO BODY POINTERS STORED AT LBOD	GENI 138
C	GENI 139
20 LTRIP=NDQ-3*NTRIP+1	GENI 140
LSCAL=LTRIP-NSCAL	GENI 141
LBOT=LSCAL	GENI 142
L=LAR	GENI 143
LBODY=L+1	GENI 144
LDATA=LBODY+3*(NBODY+NRPP)	GENI 145
LBOD=LDATA	GENI 146
C	GENI 147
C IRIPLTS	GENI 148
C	GENI 149
IF(NTRIP.EQ.0)GOTO 30	GENI 150
WRITE (6,907)	GENI 151
DO 21 I=1,NTRIP	GENI 152
I1=LTRIP+3*(I-1)	GENI 153
I2=I1+2	GENI 154
READ(5,908)(ASTER(K),K=I1,I2)	GENI 155
WRITE (6,909)(I,(ASTER(K),K=I1,I2))	GENI 156
21 CONTINUE	GENI 157
C	GENI 158
C SCALARS	GENI 159
C	GENI 160
30 IF(NSCAL.EQ.0)GOTO 50	GENI 161
I1=LSCAL	GENI 162
I2=I1+NSCAL-1	GENI 163
WRITE (6,910)	GENI 164
DO 31 I=I1,I2	GENI 165
J=I-I1+1	GENI 166
READ(5,908)ASTER(I)	GENI 167
WRITE (6,909)J,ASTER(I)	GENI 168
31 CONTINUE	GENI 169
C	GENI 170
C READ AND PROCESS BODIES	GENI 171
C	GENI 172
50 WRITE (6,911)	GENI 173
C	GENI 174
C LOOP TO PROCESS SOLIDS	GENI 175
C	GENI 176
DO 370 N=1,NBODY	GENI 177
NN=N+NRPP	GENI 178
LS1=0	GENI 179
READ(5,912) IC(1),IC(2),IC(3),ITYPE,IC(4),(FX(K),K=1,6)	GENI 180
DO 51 I=1,11	GENI 181

IF(ITYPE.EQ.ITY(1))GOTO 52	GENI 182
51 CONTINUE	GENI 183
WRITE (6,913)ITYPE	GENI 184
STOP	GENI 185
52 ITYPE=I	GENI 186
NBOD(I)=NBOD(I)+1	GENI 187
K=LBODY+3*(NRPP+N-1)	GENI 188
MASTER(K)=ITYPE*I15+LDATA	GENI 189
IF(IC(1).NE.IBL)GOTO 200	GENI 190
C	GENI 191
C BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	GENI 192
GOTO(101,103,102,101,101,102,101,100,104,101,100),ITYPE	GENI 193
100 WRITE (6,939)ITY(ITYPE)	GENI 194
STOP	GENI 195
101 LL=4	GENI 196
GOTO 110	GENI 197
102 LE=3	GENI 198
GOTO 110	GENI 199
103 LE=2	GENI 200
GOTO 110	GENI 201
104 LE=7	GENI 202
110 CALL CONVRT(FX,IX,LE)	GENI 203
WRITE (6,914)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(IX(J),J=1,LE)	GENI 204
LT=LTRIP-3	GENI 205
J1=IX(1)	GENI 206
J2=IX(2)	GENI 207
J3=IX(3)	GENI 208
J4=IX(4)	GENI 209
J5=IX(5)	GENI 210
J6=IX(6)	GENI 211
J7=IX(7)	GENI 212
C BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	GENI 213
GOTO(120,130,140,120,150,160,120,100,170,150,100),ITYPE	GENI 214
C BOX REC RAW	GENI 215
120 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LT+3*J3	GENI 216
MASTER(LDATA+1)=LT+3*J4	GENI 217
LDATA=LDATA+2	GENI 218
GOTO 360	GENI 219
C SPH	GENI 220
130 MASTER(LDATA)=(LT+3*J1)*I15+LSCAL+J2-1+I30	GENI 221
LDATA=LDATA+1	GENI 222
GOTO 360	GENI 223
C RCC	GENI 224
140 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LSCAL+J3-1	GENI 225
LDATA=LDATA+1	GENI 226
GOTO 360	GENI 227
C TRC TOR	GENI 228
150 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LSCAL+J3-1	GENI 229
MASTER(LDATA+1)=LSCAL+J4-1	GENI 230
LDATA=LDATA+2	GENI 231
GOTO 360	GENI 232
C ELL	GENI 233
160 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LSCAL+J3-1	GENI 234
LDATA=LDATA+1	GENI 235
GOTO 360	GENI 236
C TEC	GENI 237
170 MASTER(LDATA)=(LT+3*J1)*I30+(LT+3*J2)*I15+LT+3*J3	GENI 238
MASTER(LDATA+1)=(LT+3*J4)*I30+(LSCAL+J5-1)*I15+LSCAL+J6-1	GENI 239
MASTER(LDATA+2)=LSCAL+J7-1	GENI 240
LDATA=LDATA+3	GENI 241

C	GOTO 360	GENI 242
C		GENI 243
C	NOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	GENI 244
200	GOTO(201,220,202,201,203,202,201,230,204,203,240), ITYPE	GENI 245
201	LE=12	GENI 246
	GOTO 210	GENI 247
202	LL=7	GENI 248
	GOTO 210	GENI 249
203	LE=8	GENI 250
	GOTO 210	GENI 251
204	LL=13	GENI 252
210	WRITE (6,915)NN, IC(1), IC(2), IC(3), ITY(ITYPE), IC(4), (FX(J), J=1,6)	GENI 253
	READ(5,940) (FX(J), J=7, LE)	GENI 254
	WRITE (6,916) (FX(J), J=7, LE)	GENI 255
C	NOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	GENI 256
	GOTO(290,300,300,290,285,270,290,300,260,250,300), ITYPE	GENI 257
C	SPH	GENI 258
220	WRITE (6,915)NN, IC(1), IC(2), IC(3), ITY(ITYPE), IC(4), (FX(J), J=1,4)	GENI 259
	GOTO 300	GENI 260
C	ARB / R1 / V1 / 6 PER ARB	GENI 261
230	WRITE (6,915)NN, IC(1), IC(2), IC(3), ITY(ITYPE), IC(4), (FX(J), J=1,6)	GENI 262
	CALL ALBERT(FX, LBOT, NDU, LS1)	GENI 263
	GOTO 360	GENI 264
C	ARS / / LDATA /	GENI 265
240	CALL ARIN(LBOT, LDATA, MASTER, ASTER, IWH)	GENI 266
	GOTO 360	GENI 267
C	TOR CONVERT TO UNIT VECTOR	GENI 268
250	TI(1)=FX(4)	GENI 269
	TI(2)=FX(5)	GENI 270
	TI(3)=FX(6)	GENI 271
	CALL UNIT(TI)	GENI 272
	FX(4)=TI(1)	GENI 273
	FX(5)=TI(2)	GENI 274
	FX(6)=TI(3)	GENI 275
	IF(FX(7).GE.FX(8))GOTO 280	GENI 276
	WRITE (6,943)	GENI 277
	IERR=IERR+1	GENI 278
	GOTO 280	GENI 279
C	TEC	GENI 280
260	FX(13)=FX(13)	GENI 281
	LE=15	GENI 282
	TI1(1)=FX(7)	GENI 283
	TI1(2)=FX(8)	GENI 284
	TI1(3)=FX(9)	GENI 285
	TI2(1)=FX(10)	GENI 286
	TI2(2)=FX(11)	GENI 287
	TI2(3)=FX(12)	GENI 288
	IF(ABS(DOT(TI1, TI2)).LE.0.01) GOTO 265	GENI 289
	WRITE (6,942)NN, TI1, TI2	GENI 290
	IERR=IERR+1	GENI 291
C	SEMI MAJOR AXIS FX(13)	GENI 292
265	FX(13)=SQRT(DOT(TI1, TI1))	GENI 293
	CALL UNIT(TI1)	GENI 294
	FX(10)=TI1(1)	GENI 295
	FX(11)=TI1(2)	GENI 296
	FX(12)=TI1(3)	GENI 297
C	SEMI MINOR AXIS FX(14)	GENI 298
	FX(14)=SQRT(DOT(TI2, TI2))	GENI 299
C	NORMAL HEIGHT VECTOR	GENI 300
	CALL CROSS(TI, TI1, TI2)	GENI 301

CALL UNIT(TT)	GENI 302
HON=FX(4)*TT(1)+FX(5)*TT(2)+FX(6)*TT(3)	GENI 303
IF(HON)267,266,268	GENI 304
266 WRITE(6,948)	GENI 305
IERR=IERR+1	GENI 306
GOTO 268	GENI 307
267 TT(1)=-TT(1)	GENI 308
TT(2)=-TT(2)	GENI 309
TT(3)=-TT(3)	GENI 310
268 FX(7)=TT(1)	GENI 311
FX(8)=TT(2)	GENI 312
FX(9)=TT(3)	GENI 313
GOTO 280	GENI 314
C FLL COMPUTE FOCI	GENI 315
270 IF(IC(4).EQ.1BL)GOTO 300	GENI 316
ASQ=FX(4)*FX(4)+FX(5)*FX(5)+FX(6)*FX(6)	GENI 317
C=SQR(ASQ-FX(7)*FX(7))	GENI 318
A=SQR(ASQ)	GENI 319
FX(7)=A+A	GENI 320
C X,Y,Z COMPONENTS OF FOCI	GENI 321
CX=C*FX(4)/A	GENI 322
CY=C*FX(5)/A	GENI 323
CZ=C*FX(6)/A	GENI 324
C VERTEX + AND - X,Y,Z COMPONENTS GIVE THE 2 FOCI	GENI 325
FX(4)=FX(1)+CX	GENI 326
FX(5)=FX(2)+CY	GENI 327
FX(6)=FX(3)+CZ	GENI 328
FX(1)=FX(1)-CX	GENI 329
FX(2)=FX(2)-CY	GENI 330
FX(3)=FX(3)-CZ	GENI 331
C PRINT NEW INPUT	GENI 332
280 WRITE(6,915)NN,IC(1),IC(2),IC(3),ITY(ITYPE),IC(4),(FX(J),J=1,6)	GENI 333
WRITE(6,916)(FX(J),J=7,LE)	GENI 334
GOTO 300	GENI 335
C IRC CHECK R1.NE.R2	GENI 336
285 IF(FX(7).NE.FX(8))GOTO 300	GENI 337
WRITE(6,944)	GENI 338
IERR=IERR+1	GENI 339
GOTO 300	GENI 340
C BOX RAW REC CHECK IF VECTORS ARE PERPENDICULAR	GENI 341
290 IF(ABS(FX(4)*FX(7)+FX(5)*FX(8)+FX(6)*FX(9)).LE.0.01)GOTO 291	GENI 342
WRITE(6,942)NN,(FX(J),J=4,9)	GENI 343
IERR=IERR+1	GENI 344
291 IF(ABS(FX(4)*FX(10)+FX(5)*FX(11)+FX(6)*FX(12)).LE.0.01)GOTO 292	GENI 345
WRITE(6,942)NN,FX(4),FX(5),FX(6),FX(10),FX(11),FX(12)	GENI 346
IERR=IERR+1	GENI 347
292 IF(ABS(FX(7)*FX(10)+FX(8)*FX(11)+FX(9)*FX(12)).LE.0.01)GOTO 300	GENI 348
WRITE(6,942)NN,(FX(J),J=7,12)	GENI 349
IERR=IERR+1	GENI 350
C	GENI 351
C BOX SPH RCC RLC IRC ELL RAW ARB TEC IOR ARS	GENI 352
300 GOTO(310,320,330,310,340,330,310,230,350,340,240),ITYPE	GENI 353
C BOX REC RAW / V1 / V2 /	GENI 354
C / V2 / V3 /	GENI 355
310 CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)	GENI 356
MASTER(LDATA)=IWH*115	GENI 357
CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)	GENI 358
MASTER(LDATA)=MASTER(LDATA)+IWH	GENI 359
CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(8),FX(9),LBOT,LDATA,NDQ,LS1)	GENI 360
MASTER(LDATA+1)=IWH*115	GENI 361

	CALL SEE3(IWH,ASTER,MASTER,FX(10),FX(11),FX(12),	GENI 362
	1 LBOT,LDATA,NDQ,LS1)	GENI 363
	MASTER(LDATA+1)=MASTER(LDATA+1)+IWH	GENI 364
	LDATA=LDATA+2	GENI 365
	GO TO 360	GENI 366
C	SPH / V1 / R1 /	GENI 367
320	CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)	GENI 368
	MASTER(LDATA)=IWH*115	GENI 369
	LS1=1	GENI 370
	CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(4),FX(4),LBOT,LDATA,NDQ,LS1)	GENI 371
	LS1=0	GENI 372
	MASTER(LDATA)=MASTER(LDATA)+IWH	GENI 373
	LDATA=LDATA+1	GENI 374
	GO TO 360	GENI 375
C	RCC FLL / V1 / V2 /	GENI 376
C	/ R1 /	GENI 377
330	CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)	GENI 378
	MASTER(LDATA)=IWH*115	GENI 379
	CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)	GENI 380
	MASTER(LDATA)=MASTER(LDATA)+IWH	GENI 381
	LS1=1	GENI 382
	CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(7),FX(7),LBOT,LDATA,NDQ,LS1)	GENI 383
	LS1=0	GENI 384
	MASTER(LDATA+1)=IWH	GENI 385
	LDATA=LDATA+2	GENI 386
	GO TO 360	GENI 387
C	IRC TOR / V1 / V2 /	GENI 388
C	/ R1 / R2 /	GENI 389
340	CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)	GENI 390
	MASTER(LDATA)=IWH*115	GENI 391
	CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)	GENI 392
	MASTER(LDATA)=MASTER(LDATA)+IWH	GENI 393
	LS1=1	GENI 394
	CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(7),FX(7),LBOT,LDATA,NDQ,LS1)	GENI 395
	MASTER(LDATA+1)=IWH*115	GENI 396
	CALL SEE3(IWH,ASTER,MASTER,FX(8),FX(8),FX(8),LBOT,LDATA,NDQ,LS1)	GENI 397
	LS1=0	GENI 398
	MASTER(LDATA+1)=MASTER(LDATA+1)+IWH	GENI 399
	LDATA=LDATA+2	GENI 400
	GO TO 360	GENI 401
C	TEC / V1 / V2 /	GENI 402
C	/ V3 / V4 /	GENI 403
C	/ R1 / R2 /	GENI 404
C	/ R3 /	GENI 405
350	CALL SEE3(IWH,ASTER,MASTER,FX(1),FX(2),FX(3),LBOT,LDATA,NDQ,LS1)	GENI 406
	MASTER(LDATA)=IWH*115	GENI 407
	CALL SEE3(IWH,ASTER,MASTER,FX(4),FX(5),FX(6),LBOT,LDATA,NDQ,LS1)	GENI 408
	MASTER(LDATA)=MASTER(LDATA)+IWH	GENI 409
	CALL SEE3(IWH,ASTER,MASTER,FX(7),FX(8),FX(9),LBOT,LDATA,NDQ,LS1)	GENI 410
	MASTER(LDATA+1)=IWH*115	GENI 411
	CALL SEE3(IWH,ASTER,MASTER,FX(10),FX(11),FX(12),	GENI 412
	1 LBOT,LDATA,NDQ,LS1)	GENI 413
	MASTER(LDATA+1)=MASTER(LDATA+1)+IWH	GENI 414
	LS1=1	GENI 415
	CALL SEE3(IWH,ASTER,MASTER,FX(13),FX(13),FX(13),	GENI 416
	1 LBOT,LDATA,NDQ,LS1)	GENI 417
	MASTER(LDATA+2)=IWH*115	GENI 418
	CALL SEE3(IWH,ASTER,MASTER,FX(14),FX(14),FX(14),	GENI 419
	1 LBOT,LDATA,NDQ,LS1)	GENI 420
	MASTER(LDATA+2)=MASTER(LDATA+2)+IWH	GENI 421

CALL SEE3(IWH,ASTER,MASTER,FX(15),FX(15),FX(15),	GENI 422
1 LBOT,LDATA,NDQ,LS1)	GENI 423
LS1=0	GENI 424
MASTER(LDATA+3)=IWH	GENI 425
LDATA=LDATA+4	GENI 426
C CHECK IF ANY MORE ROOM FOR SOLID DATA	GENI 427
360 IF(LDATA.LT.NDQ)GOTO 370	GENI 428
WRITE (6,917)LDATA,LBOT,NDQ	GENI 429
STOP	GENI 430
370 CONTINUE	GENI 431
WRITE (6,918)	GENI 432
WRITE(6,947)IFY,NBOD	GENI 433
WRITE (6,945)LBASE,LRPPD,LABUT,LBODY,LBOD,LDATA,LBOT,LSCAL,LTRIP,NBOD	GENI 434
ADD	GENI 435
C	GENI 436
C TRANSFER ASTER(LBOT - NDQ) TO ASTER(LDATA - LDATA+LSUB)	GENI 437
C	GENI 438
LD=LDATA-1	GENI 439
LSUB=LBOT-LD-1	GENI 440
DO 375 I=LBOT,NDQ	GENI 441
ASTER(LDATA)=ASTER(I)	GENI 442
LDATA=LDATA+1	GENI 443
375 CONTINUE	GENI 444
C UNPACK POINTERS AND ADJUST FOR TRANSFER	GENI 445
K=LBODY+3*(NRPP+NBODY)	GENI 446
DO 390 I=K,LD	GENI 447
CALL UN2(I,I1,I2)	GENI 448
IF(I1.NE.0)I1=I1-LSUB	GENI 449
IF(I2.NE.0)I2=I2-LSUB	GENI 450
MASTER(I)=I1*I15+I2	GENI 451
390 CONTINUE	GENI 452
C	GENI 453
C REGION STORAGE	GENI 454
C LREGD / LOC BODY LIST / NUM OF BODIES /	GENI 455
C LDATA / OPERATOR / BODY NUM /	GENI 456
C	GENI 457
C	GENI 458
WRITE (6,920)	GENI 459
N=0	GENI 460
J=0	GENI 461
LREGD=LDATA	GENI 462
LDATA=LDATA+NRMAX	GENI 463
LREGL=LDATA	GENI 464
C	GENI 465
C READ REGION	GENI 466
C	GENI 467
400 READ(5,921)IR,(IA(I),IN(I),I=1,9)	GENI 468
C CHECK VALIDITY OF REGION DATA	GENI 469
DO 410 I=1,9	GENI 470
IF(IABS(IN(I)).LT.NBODY+NRPP)GOTO 410	GENI 471
WRITE (6,922)IR,I	GENI 472
J=J+1	GENI 473
410 CONTINUE	GENI 474
C	GENI 475
C STORE REGION DATA	GENI 476
C	GENI 477
IF(IR)440,420,421	GENI 478
420 WRITE (6,923)(IA(I),IN(I),I=1,9)	GENI 479
GOTO 430	GENI 480
421 N=N+1	GENI 481

WRITE (6,924)IR,(IA(I),IN(I),I=1,9)	GENI 482
M=LRFCD+N-1	GENI 483
MASTER(M)=LDATA*115	GENI 484
C CHECK OPERATOR	GENI 485
430 DO 435 I=1,9	GENI 486
DO 431 K=1,8	GENI 487
IF(IA(I).EQ.IAA(K))GOTO 432	GENI 488
431 CONTINUE	GENI 489
WRITE (6,925)IA(I),I	GENI 490
STOP	GENI 491
432 IA(I)=IAM(K)	GENI 492
IF(IN(I))433,400,434	GENI 493
433 IA(I)=4+IA(I)	GENI 494
IN(I)=-IN(I)	GENI 495
434 MASTER(LDATA)=IA(I)*115+IN(I)	GENI 496
LDATA=LDATA+1	GENI 497
MASTER(M)=MASTER(M)+1	GENI 498
IF(LDATA.LT.NLC)GOTO 435	GENI 499
WRITE (6,927)LDATA,NLC	GENI 500
STOP	GENI 501
435 CONTINUE	GENI 502
GOTO 400	GENI 503
L	GENI 504
C END REGION READ	GENI 505
C	GENI 506
440 IF(M.GE.NRMAX)GOTO 441	GENI 507
WRITE (6,926)IR	GENI 508
STOP	GENI 509
441 IF(J.LF.C)GOTO 442	GENI 510
WRITE (6,941)	GENI 511
STOP	GENI 512
442 WRITE (6,928)	GENI 513
C	GENI 514
C IF(IRCHEK.NE.C)TEST REGION DATA	GENI 515
C (ERROR IF POINT CAN BE IN MORE THAN 1 REGION)	GENI 516
C	GENI 517
IF(IRCHEK.EQ.NO)GOTO 500	GENI 518
WRITE (6,937)	GENI 519
LL=0	GENI 520
MIS=0	GENI 521
C	GENI 522
DO 456 I=1,NRMAX	GENI 523
JJ=I+1	GENI 524
DO 455 J=JJ,NRMAX	GENI 525
KRI=LREGD+I-1	GENI 526
CALL UN2(KRI,LOC I,NUMI)	GENI 527
KRJ=LREGD+J-1	GENI 528
CALL UN2(KRJ,LOC J,NUMJ)	GENI 529
IF(NUMI.GE.NUMJ)GOTO 450	GENI 530
IO=NUMI	GENI 531
II=NUMJ	GENI 532
GOTO 451	GENI 533
450 IO=NUMJ	GENI 534
II=NUMI	GENI 535
L=LOC I	GENI 536
LOC I=LOC J	GENI 537
LOC J=L	GENI 538
C	GENI 539
451 DO 453 KO=1,10	GENI 540
KLK=LOC I+KO-1	GENI 541

CALL UN2(KLK,IOP0,NB0)	GENI 542
DO 452 KI=1,I1	GENI 543
KLK=LOCJ+KI-1	GENI 544
CALL UN2(KLK,IOP1,NB1)	GENI 545
IF(IOP0.NE.IOP1)GOTO 452	GENI 546
IF(NB0.NE.NB1)GOTO 452	GENI 547
MIS=MIS+1	GENI 548
GOTO 453	GENI 549
452 CONTINUE	GENI 550
453 CONTINUE	GENI 551
IF(MIS.NE.I1)GOTO 454	GENI 552
WRITE (6,929)J,I	GENI 553
LL=LL+1	GENI 554
454 MIS=0	GENI 555
455 CONTINUE	GENI 556
WRITE (6,930)I	GENI 557
456 CONTINUE	GENI 558
IF(LL.GT.0)STOP	GENI 559
WRITE (6,937)	GENI 560
C	GENI 561
C IS=+1 ENTERING TABLE STORED BY I15	GENI 562
C WHICH REGIONS (J) A RAY MIGHT BE IN IF IT	GENI 563
C ENTERS A GIVEN BODY (I)	GENI 564
C IS=-1 LEAVING TABLE STORED BY I	GENI 565
C WHICH REGIONS (J) A RAY MIGHT GO INTO IF IT	GENI 566
C LEAVES A GIVEN BODY (I)	GENI 567
C	GENI 568
500 IS=-1	GENI 569
NN=NBODY+NRPP	GENI 570
LENLV=LDATA	GENI 571
DO 590 MMM=1,2	GENI 572
DO 580 I=1,NN	GENI 573
M=LBODY+3*(I-1)	GENI 574
IF(IS.GE.0)GO TO 510	GENI 575
MASTER(M+1)=MASTER(M+1)+LDATA	GENI 576
GO TO 520	GENI 577
510 MASTER(M+1)=MASTER(M+1)+LCAIA*I15	GENI 578
C	GENI 579
520 DO 570 J=1,NRMAX	GENI 580
ITEMP=LREGD+J-1	GENI 581
CALL UN2(ITEMP,LOC,NC)	GENI 582
CALL UN2(LOC,IOP,DUM)	GENI 583
DO 560 N=1,NC	GENI 584
MM=LOC+N-1	GENI 585
CALL UN2(MM,IOPER,NUM)	GENI 586
IF(NUM.NE.I)GOTO 560	GENI 587
IF(IOP.EQ.1.OR.IOP.EQ.5)GOTO 540	GENI 588
IF(IOPER.GT.4)GOTO 530	GENI 589
IF(IS-1)560,550,560	GENI 590
530 IF(IS+1)560,551,560	GENI 591
540 IF(IS.LT.0)GOTO 551	GENI 592
550 MASTER(M+2)=MASTER(M+2)+I15	GENI 593
GO TO 552	GENI 594
551 MASTER(M+2)=MASTER(M+2)+1	GENI 595
552 MASTER(LDATA)=J	GENI 596
LDATA=LDATA+1	GENI 597
IF(LDATA.LT.NDQ)GOTO 570	GENI 598
WRITE (6,931)LDATA,NDQ,MMM,I	GENI 599
STOP	GENI 600
560 CONTINUE	GENI 601

570	CONTINUE	GENI 602
580	CONTINUE	GENI 603
	WRITE (6,938)MMM	GENI 604
	IS=IS+2	GENI 605
590	CONTINUE	GENI 606
C	RIN STORAGE ROUT STORAGE G1 TEMP STORAGE	GENI 607
	L1=LDATA-1	GENI 608
	NN=NRPP+NBODY	GENI 609
	LKIN=LDATA+1	GENI 610
	LROT=LRIN+NA	GENI 611
	LIO=LROT+NN	GENI 612
	LEGEOM=LIO+NN	GENI 613
	WRITE (6,932)LEGEOM	GENI 614
	WRITE (6,919)LREGD,LREGL,LENLV,LRIN,LROT,LIO,LEGEOM	GENI 615
C		GENI 616
C	PRINT ENTERING AND LEAVING TABLE	GENI 617
C		GENI 618
	IF (IFNTLV.EQ.40)RETURN	GENI 619
	WRITE (6,946)	GENI 620
	NBNR=NBODY+NRPP	GENI 621
C		GENI 622
	DO 600 J=1,NBNR	GENI 623
	LOC=LBODY+3*(N-1)	GENI 624
	LOC=LOC+1	GENI 625
	CALL UN2(LOC,LENT,LEAV)	GENI 626
	LOC=LOC+1	GENI 627
	CALL UN2(LOC,NENT,NEAV)	GENI 628
	J1=LENT	GENI 629
	J2=LENT+NENT-1	GENI 630
	WRITE (6,933)N,J1,J2,(MASTER(K),K=J1,J2)	GENI 631
	J1=LEAV	GENI 632
	J2=LEAV+NEAV-1	GENI 633
	WRITE (6,934)N,J1,J2,(MASTER(K),K=J1,J2)	GENI 634
600	CONTINUE	GENI 635
C		GENI 636
C	MASTER-ASTER ARRAY OUTPUT	GENI 637
C		GENI 638
	IF (IPRIN.EQ.0)RETURN	GENI 639
	WRITE (6,935)	GENI 640
C		GENI 641
	DO 620 K=LBASE,L1,3	GENI 642
	IK=K	GENI 643
	IK2=K+2	GENI 644
	M=0	GENI 645
	DO 610 I=IK,IK2	GENI 646
	M=M+1	GENI 647
	CALL UN2(I,I1,I2)	GENI 648
	NO1(M)=I1	GENI 649
	NO2(M)=I2	GENI 650
	O4(M)=ASTER(I)	GENI 651
	NOO(M)=I	GENI 652
610	CONTINUE	GENI 653
	WRITE (6,936)(NOO(L),NO1(L),NO2(L),O4(L),L=1,3)	GENI 654
620	CONTINUE	GENI 655
	RETURN	GENI 656
	END	GENI 657
C		GENI 658
C		GENI 659
	SUBROUTINE RPPIN(LAR)	**** 15
	DIMENSION MASTER(30000),X(6)	RPPIN 2

COMMON ASTER(30000)	RPPIN 3
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERK,DIST	RPPIN 4
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	RPPIN 5
1 LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	RPPIN 6
COMMON/R&PP/LRPPD,LABUT	RPPIN 7
EQUIVALENCE(MASTER,ASTER)	RPPIN 8
C	RPPIN 9
910 FORMAT(6F12.6)	RPPIN 10
920 FORMAT(18,17X,6F12.5)	RPPIN 11
930 FORMAT(1H0,27HERROR IN DESCRIPTION OF RPP,15,5X,10HMIN.GE.MAX)	RPPIN 12
940 FORMAT(1H0,27HERROR IN DESCRIPTION OF RPP,7X,110,10X,110)	RPPIN 13
950 FORMAT(10X,7HSURFACE,15,8X,2E20.6)	RPPIN 14
C	RPPIN 15
C N IS RPP NUMBER J IS SURFACE NUMBER	RPPIN 16
C	RPPIN 17
C MASTER-ASTER STORAGE FOR RPP	RPPIN 18
C	RPPIN 19
C LBASE - RPP POINTERS RESERVE 12 WORDS/RPP	RPPIN 20
C / I / J /	RPPIN 21
C / / K /	RPPIN 22
C I (POINTER TO LIST OF ABUTING RPP'S)	RPPIN 23
C J (NUM OF RPP'S THAT ABUT THIS SURFACE)	RPPIN 24
C K (POINTER TO MIN OR MAX CORRESPONDING	RPPIN 25
C TO THIS SURFACE)	RPPIN 26
C	RPPIN 27
C LRPPD - RPP DATA STARTING AT LBASE + 12 * NRPP	RPPIN 28
C MIN OR MAX K POINTS HERE	RPPIN 29
C	RPPIN 30
C LABUT TO LBODY-1	RPPIN 31
C LIST OF ABUTING RPP'S PACKED 1 OR 2/WORD	RPPIN 32
C 1 POINTS HERE / 1 / 2 /	RPPIN 33
C J CONTAINS NUMBER IN LIST	RPPIN 34
C	RPPIN 35
C IERR=0	RPPIN 36
C N=1	RPPIN 37
C I=LBASE+12*NRPP	RPPIN 38
C LRPPD=I	RPPIN 39
10 READ(5,910)(X(J),J=1,6)	RPPIN 40
WRITE (6,920)N,(X(J),J=1,6)	RPPIN 41
DO 20 J=1,6,2	RPPIN 42
IF(X(J).LT.X(J+1))GOTO 20	RPPIN 43
WRITE (6,930)N	RPPIN 44
STOP	RPPIN 45
20 CONTINUE	RPPIN 46
C	RPPIN 47
C STORE MIN AND MAX BEGINNING AT LBASE + 12 * NRPP	RPPIN 48
C	RPPIN 49
C DO 33 J=1,6	RPPIN 50
II=LBASE+12*NRPP	RPPIN 51
L=LBASE+12*(N-1)+2*(J-1)	RPPIN 52
30 IF(II.LT.I)GOTO 31	RPPIN 53
ASTER(II)=X(J)	RPPIN 54
MASTER(L+1)=I	RPPIN 55
I=I+1	RPPIN 56
GOTO 33	RPPIN 57
C CHECK FOR DUPLICATION	RPPIN 58
31 IF(X(J).EQ.ASTER(II))GOTO 32	RPPIN 59
II=II+1	RPPIN 60
GOTO 30	RPPIN 61
32 MASTER(L+1)=II	RPPIN 62

```

33 CONTINUE
  IF(N.GE.NRPP)GOTO 40
  N=N+1
  GOTO 10
C
40 LABUT=I
  LAST=I-1
  L=LAST
C
  SEARCH FOR ABUTING RPP'S TO SURFACE
C
  DO 57 I=1,NRPP
  DO 57 N=1,6
  LL=0
  M=1
  K=LBASE+12*(I-1)+2*(N-1)
  MASTER(K)=(L+1)*115+MASTER(K)
  NC=3*N-1-4*(N/2)
  DO 56 J=1,NRPP
  IF(I.FQ.J)GOTO 56
  IF(S(I,I).NE.S(J,NC))GOTO 56
C
  DO 53 K=1,3
  NN=N+NC
  K41=4*K-1
  IF(NN.FQ.K41)GOTO 53
  K2=2*K
  K21=K2-1
  IF(S(I,K21).GT.S(J,K21))GOTO 50
  IF(S(J,K21).LT.S(I,K2 ))GOTO 53
50 IF(S(I,K21).GL.S(J,K2 ))GOTO 51
  IF(S(J,K2 ).LL.S(I,K2 ))GOTO 53
51 IF(S(I,K2 ).GT.S(J,K2 ))GOTO 56
  IF(S(I,K21).LT.S(J,K21))GOTO 56
53 CONTINUE
  M=-M
  IF(M.LT.0)GOTO 54
  MASTER(L)=MASTER(L)+J
  GOTO 55
54 L=L+1
  MASTER(L)=J*115
55 LL=LL+1
56 CONTINUE
  K=LBASE+12*(I-1)+2*(N-1)
  MASTER(K)=MASTER(K)+LL
57 CONTINUE
C
  TEST VALIDITY OF RPP DATA
C
  IF(NRPP.LE.1)GOTO 63
C
  DO 62 J=1,6
  NRPP1=NRPP-1
  DO 61 I=1,NRPP1
  JJ=LBASE+12*(I-1)+2*(J-1)
  CALL UN2(JJ,I0UM,I2)
  I3=MASTER(JJ+1)
  IF(I2.NE.0)GOTO 61
  II=I+1
  DO 60 K=II,NRPP

```

```

RPPIN 63
RPPIN 64
RPPIN 65
RPPIN 66
RPPIN 67
RPPIN 68
RPPIN 69
RPPIN 70
RPPIN 71
RPPIN 72
RPPIN 73
RPPIN 74
RPPIN 75
RPPIN 76
RPPIN 77
RPPIN 78
RPPIN 79
RPPIN 80
RPPIN 81
RPPIN 82
RPPIN 83
RPPIN 84
RPPIN 85
RPPIN 86
RPPIN 87
RPPIN 88
RPPIN 89
RPPIN 90
RPPIN 91
RPPIN 92
RPPIN 93
RPPIN 94
RPPIN 95
RPPIN 96
RPPIN 97
RPPIN 98
RPPIN 99
RPPIN100
RPPIN101
RPPIN102
RPPIN103
RPPIN104
RPPIN105
RPPIN106
RPPIN107
RPPIN108
RPPIN109
RPPIN110
RPPIN111
RPPIN112
RPPIN113
RPPIN114
RPPIN115
RPPIN116
RPPIN117
RPPIN118
RPPIN119
RPPIN120
RPPIN121
RPPIN122

```

```

      XK=LBASE+12*(K-1)+2*(J-1)
      CALL UN2(KK, IDUM, I5)
      I6=MASTER(KK+1)
      IF(I5.NE.0)GOTO 60
      IF(I3.EQ.I6)GOTO 60
      IERR=IERR+1
      WRITE (6,940)I,K
      WRITE (6,950)J,ASTER(I3),ASTER(I6)
60  CONTINUE
      GOTO 62
61  CONTINUE
62  CONTINUE
63  LAR=L
      RETURN
      END

```

C
C

```

      SUBROUTINE ALBERT(FX, LBOT, NDQ, LS1)
      DIMENSION MASTER(30000), IA(6,4), AA(8,3), F(4), FX(6)
      COMMON ASTER(30000)
      COMMON/UNCSEM/NRPP, NTRIP, NSCAL, NBDY, NRMAX, LTRIP, LSCAL, LREGD,
1  LDAIA, LRIN, LROT, LIO, LOCCA, I15, I30, LBODY, NASC, KLOOP
      COMMON/UCOM/LBASE, RIN, ROUT, LRI, LRO, PINF, IERR, DIST
      EQUIVALENCE(ASTER, MASTER)

```

C

```

901  FORMAT(25X,6F12.5)
902  FORMAT(10X,6(1X,4I1))
903  FORMAT(10X,6E10.3)
904  FORMAT(25X,6(4X,4I2))
905  FORMAT(1H0,15HUNDEFINED PLANE)
906  FORMAT(15,10(E11.4))
907  FORMAT(1H0,26HFOUR POINTS NOT IN A PLANE)
908  FORMAT(1H0,25HERROR IN SIDE DESCRIPTION)
909  FORMAT(1H0,16HDEGENERATE PLANE,15)

```

C

```

      K=1
      DO 10 I=1,2
      DO 10 J=1,3
      AA(I,J)=FX(K)
      K=K+1
10  CONTINUE
      READ(5,903)((AA(I,J),J=1,3),I=3,8)
      READ(5,902)((IA(I,J),J=1,4),I=1,6)
      WRITE (6,901)((AA(I,J),J=1,3),I=3,8)
      WRITE (6,904)((IA(I,J),J=1,4),I=1,6)

```

C

```

      DO 70 I=1,6
      IX=IA(I,1)
      IY=IA(I,2)
      IZ=IA(I,3)
      X1=AA(IX,1)
      Y1=AA(IX,2)
      Z1=AA(IX,3)
      X2=AA(IY,1)
      Y2=AA(IY,2)
      Z2=AA(IY,3)
      X3=AA(IZ,1)
      Y3=AA(IZ,2)
      Z3=AA(IZ,3)
      D=X1*(Y2*Z3-Z2*Y3)-X2*(Y1*Z3-Z1*Y3)+X3*(Y1*Z2-Z1*Y2)

```

RPPIN123
RPPIN124
RPPIN125
RPPIN126
RPPIN127
RPPIN128
RPPIN129
RPPIN130
RPPIN131
RPPIN132
RPPIN133
RPPIN134
RPPIN135
RPPIN136
RPPIN137
RPPIN138
RPPIN139

****1,16
ALBERT 2
ALBERT 3
ALBERT 4
ALBERT 5
ALBERT 6
ALBERT 7
ALBERT 8
ALBERT 9
ALBERT10
ALBERT11
ALBERT12
ALBERT13
ALBERT14
ALBERT15
ALBERT16
ALBERT17
ALBERT18
ALBERT19
ALBERT20
ALBERT21
ALBERT22
ALBERT23
ALBERT24
ALBERT25
ALBERT26
ALBERT27
ALBERT28
ALBERT29
ALBERT30
ALBERT31
ALBERT32
ALBERT33
ALBERT34
ALBERT35
ALBERT36
ALBERT37
ALBERT38
ALBERT39
ALBERT40
ALBERT41
ALBERT42
ALBERT43


```

A=(-Y2*Z3+Z2*Y3+Y1*Z3-Z1*Y3-Y1*Z2+Z1*Y2)
B=(X2*Z3-Z2*X3-X1*Z3+X3*Z1+X1*Z2-Z1*X2)
C=(Y2*X3-X2*Y3-Y1*X3+X1*Y3+Y1*X2-X1*Y2)
D12=(X1-X3)**2+(Y1-Y3)**2+(Z1-Z3)**2
A2B2C2=A*A+B*B+C*C
IF(A2B2C2.NE.C.)GOTO 21
WRITE (6,909)I
D=ABS(D)
GOTO 61
21 D1210=D12*1.0E-12
IF(A2B2C2.GT.D1210)GOTO 22
WRITE (6,905)
WRITE (6,906)I,A,B,C,D,D12
IERR=IERR+1
GOTO 70
22 S=SQRT(A2B2C2)
WX=A/S
WY=B/S
WZ=C/S
IC=IA(1,4)
X4=AA(IC,1)
Y4=AA(IC,2)
Z4=AA(IC,3)
D2=(-D-(A*X4)-(B*Y4)-(C*Z4))/((A*WX)+(B*WY)+(C*WZ))
D22=D2*D2
C THE NEXT CARD BYPASSES THE 4TH POINT TEST
C IF(D22.LE.0.01)GOTO 30 & PRINT 907 & IERR=IERR+1
IF(D22.LE.1.01)GOTO 30
WRITE (6,907)
IERR=IERR+1
WRITE (6,906)I,A,B,C,D,D12,D2
GOTO 70
C
30 DO 31 K=1,4
F(K)=0.
31 CONTINUE
L=1
DO 32 J=1,8
IF(J.EQ.1X.OR.J.EQ.1Y.OR.J.EQ.1Z.OR.J.EQ.1C)GOTO 32
F(L)=A*AA(J,1)+B*AA(J,2)+C*AA(J,3)+D
L=L+1
32 CONTINUE
M=0
N=0
J=0
C
DO 44 L=1,4
IF(ABS(F(L)).LE.1.0E-6)GOTO 42
IF(F(L))41,42,43
41 M=M+1
GOTO 44
42 N=N+1
GOTO 44
43 J=J+1
44 CONTINUE
C
IF(N.EQ.0)GOTO 51
IF(M+N.EQ.4)GOTO 60
IF(J+N.EQ.4)GOTO 61
GOTO 52

```

ALBERT44
ALBERT45
ALBERT46
ALBERT47
ALBERT48
ALBERT49
ALBERT50
ALBERT51
ALBERT52
ALBERT53
ALBERT54
ALBERT55
ALBERT56
ALBERT57
ALBERT58
ALBERT59
ALBERT60
ALBERT61
ALBERT62
ALBERT63
ALBERT64
ALBERT65
ALBERT66
ALBERT67
ALBERT68
ALBERT69
ALBERT70
ALBERT71
ALBERT72
ALBERT73
ALBERT74
ALBERT75
ALBERT76
ALBERT77
ALBERT78
ALBERT79
ALBERT80
ALBERT81
ALBERT82
ALBERT83
ALBERT84
ALBERT85
ALBERT86
ALBERT87
ALBERT88
ALBERT89
ALBERT90
ALBERT91
ALBERT92
ALBERT93
ALBERT94
ALBERT95
ALBERT96
ALBERT97
ALBERT98
ALBERT99
ALBERT100
ALBERT101
ALBERT102
ALBERT103

51 IF(M.EQ.4)GOTO 60	ALBER104
IF(J.EQ.4)GOTO 61	ALBER105
52 WRITE (6,908)	ALBER106
WRITE (6,906)I,A,B,C,D,D12,D2,(F(L),L=1,4)	ALBER107
IERK=IERK+1	ALBER108
GOTO 70	ALBER109
C	ALBER110
60 A=-A	ALBER111
B=-B	ALBER112
C=-C	ALBER113
D=-D	ALBER114
61 CALL SEE3(IWH,ASTER,MASTER,A,B,C,LBOT,LDATA,NDQ,LS1)	ALBER115
MASTER(LDATA)=IWH	ALBER116
LS1=1	ALBER117
CALL SEE3(IWH,ASTER,MASTER,D,D,D,LBOT,LDATA,NDQ,LS1)	ALBER118
LS1=0	ALBER119
MASTER(LDATA)=MASTER(LDATA)+IWH*115	ALBER120
LDATA=LDATA+1	ALBER121
70 CONTINUE	ALBER122
RETURN	ALBER123
END	ALBER124
C	ALBER125
C	ALBER126
SUBROUTINE ARIN(LBOT,LDATA,MASTER,ASTER,IWH)	**** 17
DIMENSION MASTER(30000),ASTER(30000)	ARIN 2
COMMON/UNCLE/NN,IC(4)	ARIN 3
C	ARIN 4
C	ARIN 5
C	ARIN 6
SEE ARS SUBROUTINE FOR STORAGE IN MASTER-ASTER ARRAY	ARIN 7
901 FORMAT(10X,2I10)	ARIN 8
902 FORMAT(10X,6E10.3)	ARIN 9
903 FORMAI(I8,1X,3A1,2X,3HARS,2X,A4,6X,19HNUMBER OF CURVES IS,I6,	ARIN 10
1 5X,29HNUMBER OF POINTS PER CURVE IS,I6/)	ARIN 11
904 FORMAT(25X,6F12.5)	ARIN 12
C	ARIN 13
C	ARIN 14
C	ARIN 15
C	ARIN 16
MAX = NUM OF CURVES	ARIN 17
NAX = NUM OF POINTS/CURVE	ARIN 18
READ(5,901)MAX,NAX	ARIN 19
WRITE (6,903)NN,(IC(I),I=1,4),MAX,NAX	ARIN 20
LBOT=LBOT-4*MAX*NAX-92	ARIN 21
IWH=LBOT	ARIN 22
MASTER(LDATA)=IWH	ARIN 23
LDATA=LDATA+1	ARIN 24
C	ARIN 25
DO 50 M=1,MAX	ARIN 26
L1=LBOT+92+4*NAX*(M-1)	ARIN 27
L2=L1+4*NAX-1	ARIN 28
READ(5,902)(ASTER(L),ASTER(L+1),ASTER(L+2),L=L1,L2,4)	ARIN 29
WRITE (6,904)(ASTER(L),ASTER(L+1),ASTER(L+2),L=L1,L2,4)	ARIN 30
WRITE (6,904)	ARIN 31
50 CONTINUE	ARIN 32
MASTER(LBOT)=0	ARIN 33
MASTER(LBOT+1)=MAX	ARIN 34
MASTER(LBOT+2)=NAX	ARIN 35
RETURN	ARIN 36
END	**** 18
C	
C	
SUBROUTINE SEE3(IWH,ASTER,MASTER,FX,FXX,FXXX,LBOT,LDATA,NDQ,LS1)	

C	DIMENSION ASTER(30000),MASTER(30000)	SEE3	2
C	STOKES TRIPLETS AND SCALARS IN MASTER-ASTER ARRAY	SEE3	3
C		SEE3	4
C	IF(LSI.NE.0)GOTO 50	SEE3	5
C	TRIPLLTS	SEE3	6
C	IF(LBOT.GT.NDQ)GOTO 20	SEE3	7
	NDQ2=NDQ-2	SEE3	8
	DO 10 I=LBOT,NDQ2	SEE3	9
	IF(ASTER(I).NE.FX)GOTO 10	SEE3	10
	IF(ASTER(I+1).NE.FXX)GOTO 10	SEE3	11
	IF(ASTER(I+2).NE.FXXX)GOTO 10	SEE3	12
	IWH=I	SEE3	13
	RETURN	SEE3	14
10	CONTINUE	SEE3	15
20	ASTER(LBOT-1)=FXXX	SEE3	16
	ASTER(LBOT-2)=FXX	SEE3	17
	ASTER(LBOT-3)=FX	SEE3	18
	LBOT=LBOT-3	SEE3	19
	IWH=LBOT	SEE3	20
	IF(LBOT.LE.LDATA)WRITE (6,30)LBOT,LDATA	SEE3	21
	RETURN	SEE3	22
30	FORMAT(1H0,22HMEMORY OVERLAP IN SEE3,5X,5HLBOT=,110,	SEE3	23
	1 5X,6HLDATA=,110)	SEE3	24
C		SEE3	25
C	SCALARS	SEE3	26
50	DO 60 I=LBOT,NDQ	SEE3	27
	IF(ASTER(I).NE.FX)GOTO 60	SEE3	28
	IWH=I	SEE3	29
	RETURN	SEE3	30
60	CONTINUE	SEE3	31
	ASTER(LBOT-1)=FX	SEE3	32
	LBOT=LBOT-1	SEE3	33
	IWH=LBOT	SEE3	34
	RETURN	SEE3	35
	END	SEE3	36
C		SEE3	37
C		SEE3	38
C		SEE3	39
	FUNCTION S(I,N)	****	19
	DIMENSION MASTER(30000)	S	2
	COMMON ASTER(30000)	S	3
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	S	4
	EQUIVALENCE(MASTER,ASTER)	S	5
C		S	6
C	S RETRIEVES COORDINATES OF ANY OF THE 6 SIDES OF AN RPP	S	7
C	I IS RPP NUMBER N IS SURFACE NUMBER	S	8
C		S	9
	L=LBASE+12*(I-1)+2*(N-1)	S	10
	LL=MASTER(L+1)	S	11
	S=ASTER(LL)	S	12
	RETURN	S	13
	END	S	14
C		S	15
C		S	16
	SUBROUTINE CONVRT(FX,IX,LE)	****	20
	DIMENSION FX(6),IX(6)	CONVRT	2
C	LE NUMBER OF REFERENCES TO SCALARS AND TRIPLETS	CONVRT	3
C	INTLGAL PART OF FX CONVERTED TO FIXED POINT NUM IN IX(1)	CONVRT	4
C	FRACTIONAL PART OF FX CONVERTED TO FIXED POINT NUM IN IX(11)	CONVRT	5
	NFX=(LE+1)/2	CONVRT	6

```

DO 10 IFX=1,NFX
  I1=2*IFX
  I=I1-1
  IX(I)=FX(IFX)+.000001
  X=IX(I)
  IX(I1)=(FX(IFX)-X)*100000+.00001
10 CONTINUE
RETURN
END

```

C
C

```

SUBROUTINE GRID
  DIMENSION WP(3)
  COMMON/PAREM/XB(3),WB(3),IR
  COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
  COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
1  LDATA,LKIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
  COMMON/GTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,
1  ITR(200),CA,CE,SA,SE
  COMMON/CAL/NIK,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),
1  TRAVEL,SN,V,H,IVIH
  COMMON/WALT/LIRFO,NGIERR
  COMMON/HOYT/VREF,HREF
  COMMON/CELL/CELSIZ
  COMMON/CONTRL/ITESTG,IRAYSK,IENLV,IVOLUM,IWOT,ITAPE8,NO,IYES

```

C

```

901 FORMAT(8I10)
902 FORMAT(6E12.8)
903 FORMAT(1H0,2HNX,15,5X,2HNY,15,5X,7HIRSTRT,15,5X,4HIENC,15,5X,
1  6HNSTART,16,5X,4HNEND,16,5X,9HCELL SIZE,F7.2//
2  17H DATUM LINE AT Z=,F10.3,27H WITH RESPECT TO THE ORIGIN/
3  17H GROUND IS AT Z=,F10.3,27H WITH RESPECT TO THE ORIGIN/
4  17H XSHIFT IS AT X=,F10.3,27H WITH RESPECT TO THE ORIGIN/
5  17H YSHIFT IS AT Y=,F10.3,27H WITH RESPECT TO THE ORIGIN/)
904 FORMAT(1H ,7HAZIMUTH,F12.5,5X,9HELEVATION,F12.5,5X,
1  13HBACK OFF DIST,F12.5)
905 FORMAT(2E20.8,4E10.3)
906 FORMAT(52HOTHIS RAY WAS SUPPRESSED BECAUSE IT WAS BELOW GROUND)
907 FORMAT(1H0,15,15H CELLS SKIPPED)

```

C

```

READ (5,901)NX,NY,IRSTRT, IENC,NGIERR,NSTART,NEND
READ (5,902)A,E,ENGTH,ZSHIFT,GROUND
READ (5,902)XSHIFT,YSHIFT,CELSIZ
IF(IRSTRT .LE.0)IRSTRT=1
IF(CELSIZ .LE.0.)CELSIZ=4.
IF(NSTART.LE.0)NSTART=1
IF(NEND.LE.NSTART)NEND=NX*NY
IF(NGIERR.LE.0)NGIERR=25

```

C

```

WRITE (6,903)NX,NY,IRSTRT, IENC,NSTART,NEND,CELSIZ,
1  ZSHIFT,GROUND,XSHIFT,YSHIFT
IF(IWOT.EQ.IYES)WRITE(1,905)A,E,XSHIFT,YSHIFT,ZSHIFT,CELSIZ
WRITE (6,904)A,E,ENGTH
RADIAN=.017453292519943
AR=A*RADIAN
ER=E*RADIAN
WRITE (6,904)AR,FK,ENGTH
SA=SIN(AR)
CA=COS(AR)
SE=SIN(ER)

```

```

CONVRT 7
CONVRT 8
CONVRT 9
CONVRT10
CONVRT11
CONVRT12
CONVRT13
CONVRT14
CONVRT15
CONVRT16
CONVRT17
**** 21
GRID 2
GRID 3
GRID 4
GRID 5
GRID 6
GRID 7
GRID 8
GRID 9
GRID 10
GRID 11
GRID 12
GRID 13
GRID 14
GRID 15
GRID 16
GRID 17
GRID 18
GRID 19
GRID 20
GRID 21
GRID 22
GRID 23
GRID 24
GRID 25
GRID 26
GRID 27
GRID 28
GRID 29
GRID 30
GRID 31
GRID 32
GRID 33
GRID 34
GRID 35
GRID 36
GRID 37
GRID 38
GRID 39
GRID 40
GRID 41
GRID 42
GRID 43
GRID 44
GRID 45
GRID 46
GRID 47
GRID 48
GRID 49

```

	CE=CEPS(ER)	GRID 50
C		GRID 51
C	PROCESS KL CELLS IN GRID	GRID 52
C		GRID 53
	DO 40 KK=NSTART,NEND	GRID 54
	WB(1)=-CL*CA	GRID 55
	WB(2)=-CF*SA	GRID 56
	WB(3)=-SE	GRID 57
	II=((KK-1)/NX)+1	GRID 58
	J=KK-(II-1)*NX	GRID 59
C	COMPUTE COORDINATES OF GRID CELL IN GRID PLANE	GRID 60
	CELL2=.5*CELSIZ	GRID 61
	V=FLOAT((NY/2)-II)*CELSIZ +CELL2	GRID 62
	VREF=V+CELL2	GRID 63
	H=FLOAT((NX/2)- J)*CELSIZ +CELL2	GRID 64
	HREF=H+CELL2	GRID 65
	IV=RAN(-1)*10.	GRID 66
	IH=RAN(-1)*10.	GRID 67
	IVIH=10*IH+IV	GRID 68
C	COMPUTE H,V AT RANDOM POINT IN GRID CELL	GRID 69
	V=V+CELSIZ *FLOAT(IV)/10.+CELSIZ /20.	GRID 70
	H=H+CELSIZ *FLOAT(IH)/10.+CELSIZ /20.	GRID 71
C	X,Y,Z IN COORDINATE SYSTEM OF VEHICLE	GRID 72
	XBS(1)=XSHIFT-V*CA*SE-H*SA	GRID 73
	XBS(2)=YSHIFT-V*SA*SE+H*CA	GRID 74
	XPS(3)=ZSHIFT+V*CE	GRID 75
	CALL TROPIC(WP)	GRID 76
	XBS(1)=XBS(1)+WP(1)*1.0E-4	GRID 77
	XBS(2)=XPS(2)+WP(2)*1.0E-4	GRID 78
	XBS(3)=XPS(3)+WP(3)*1.0E-4	GRID 79
	XB(1)=XBS(1)-ENGTH*WB(1)	GRID 80
	XB(2)=XBS(2)-ENGTH*WB(2)	GRID 81
	XB(3)=XBS(3)-ENGTH*WB(3)	GRID 82
	IF(XB(3).GT.GROUND)GOTO 10	GRID 83
	IF(ITAPE8.EQ.IYES)WRITE (6,906)	GRID 84
	GOTO 40	GRID 85
10	DO 20 KK1=1,3	GRID 86
	XS(KK1)=XB(KK1)	GRID 87
	WS(KK1)=WB(KK1)	GRID 88
20	CONTINUE	GRID 89
	CALL TRACK	GRID 90
	IF(IERR.GE.NGIERR)RETURN	GRID 91
	IF(IRAYSK.EQ.NO)GOTO 40	GRID 92
	MSHIFT=RAN(-1)*25.	GRID 93
	WRITE (6,907)MSHIFT	GRID 94
	KK=KK+MSHIFT	GRID 95
40	CONTINUE	GRID 96
	RETURN	GRID 97
	END	GRID 98
C		GRID 99
C		GRID 100
C		GRID 101
	SUBROUTINE TRACK	**** 22
	DIMENSION XP(3),ERROR(2)	TRACK 2
	COMMON/PAREN/XB(3),WB(3),IR	TRACK 3
	COMMON/GEOM/LBASEF,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	TRACK 4
	COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	TRACK 5
1	LDATA,LRIN,LRJT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	TRACK 6
	COMMON/GTRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,	TRACK 7
1	ITR(200),CA,CE,SA,SE	TRACK 8

COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),TRAVEL,	TRACK 9
1 SN,V,H,IVIH	TRACK 10
COMMON/CONTRL/ITESTG,IRAYSK,IENTLV,IVOLUM,IWOT,ITAPE8,NO,IYES	TRACK 11
COMMON/WALT/LIRFO,NGIERR	TRACK 12
COMMON/HOYT/VREF,HREF	TRACK 13
COMMON/LSU/LSURF	TRACK 14
COMMON/CELL/CELSIZ	TRACK 15
COMMON/ERR/IERRO	TRACK 16
C	TRACK 17
901 FORMAT(F6.1,1X,F6.1,1X,I2,1X,F7.2,1X,F7.2,4(1X,I1),I3,1X,2I3,	TRACK 18
1 1X,F8.3,1X,F8.3)	TRACK 19
902 FORMAT(2(I4,F7.2,F7.2,F6.1,I3,F7.2),1X,2I3,1X,I1,I1,I2,4X,A6)	TRACK 20
903 FORMAT(31H NUMBER OF INTERSECTIONS.GT.200)	TRACK 21
904 FORMAT(//)	TRACK 22
905 FORMAT(1H0,16H0 ITEM IN CELL (,I4,1H,,I4,1H),5X,	TRACK 23
1 2HH=,F6.1,5X,2HV=,F6.1)	TRACK 24
C	TRACK 25
ERROR(2)= 6H0 ITEM	TRACK 26
DATA ERROR(1),ERROR(2)/4H ,4HITEM/	TRACK 27
I12=4096	TRACK 28
NASC=-1	TRACK 29
IR=IRSTRT	TRACK 30
L=1	TRACK 31
KHIT=0	TRACK 32
JCNT=0	TRACK 33
MSKHT=0	TRACK 34
MTARG=1	TRACK 35
MARMR=0	TRACK 36
MVOL =0	TRACK 37
C	TRACK 38
DO 10 I=1,200	TRACK 39
ITR(I)=0	TRACK 40
IR(I)=0.	TRACK 41
10 CONTINUE	TRACK 42
C	TRACK 43
C S1 IS DISTANCE THRU REGION IR	TRACK 44
C IRPRIM IS NEW REGION NUMBER	TRACK 45
C XP IS POINT OF CONTACT	TRACK 46
C	TRACK 47
20 CALL G1(S1,IRPRIM,XP)	TRACK 48
IF(IRPRIM.LT.0)RETURN	TRACK 49
IR(L)=S1	TRACK 50
KLSURF=LSURF+7	TRACK 51
LOC=LIRFO+IR-1	TRACK 52
CALL UN2(LOC,DUM,IDENT)	TRACK 53
IDENT=IDENT-1	TRACK 54
C / SURFACE NUM / BODY NUM / NEXT REGION /	TRACK 55
ITR(L)=(KLSURF*I12+NASC)*I12+IRPRIM	TRACK 56
IF(NASC.LE.NRPP)IRPRIM=0	TRACK 57
IF(IRPRIM.EQ.0)GOTO 100	TRACK 58
IR=IRPRIM	TRACK 59
KHIT=KHIT+1	TRACK 60
IF(L.GT.1)GOTO 40	TRACK 61
SUM=0.	TRACK 62
DO 30 I=1,3	TRACK 63
SUM=SUM+WS(I)*XP(I)	TRACK 64
30 CONTINUE	TRACK 65
D1=-SUM	TRACK 66
GOTO 60	TRACK 67
C	TRACK 68

C	CHECK IDENT CODE	0 NONE	TRACK 69	
C	10=SKIRT	20=ARMOR	30=TARGET	TRACK 70
C	SPACE CODES	1 EXTERIOR VOLUME		TRACK 71
C		-1,2-9,11-19,21-29,.....,91-99 INTERIOR VOLUME		TRACK 72
C				TRACK 73
	40	IF(IDENT.EQ.0)GOTO 60	TRACK 74	
		IF(IDENT-(IDENT/10)*10.EQ.0)GOTO 50	TRACK 75	
		KHIT=KHIT-1	TRACK 76	
		IF(IDENT.NE.1)MVOL=1	TRACK 77	
		GOTO 60	TRACK 78	
C			TRACK 79	
	50	IF(IDENT.LQ.20)MARMR=1	TRACK 80	
		IF(IDENT.EQ.30)MTARG=1	TRACK 81	
		IF(IDENT.EQ.10)MSKRT=1	TRACK 82	
	60	L=L+1	TRACK 83	
		IF(L.LE.200)GOTO 20	TRACK 84	
		WRITE (6,903)	TRACK 85	
		STOP	TRACK 86	
C			TRACK 87	
C	END OF RAY	PRINT RESULTS	TRACK 88	
C			TRACK 89	
	100	IF(L.EQ.1)RETURN	TRACK 90	
		IF(IIAPLS.EQ.NO.AND.IWOT.EQ.NO)RETURN	TRACK 91	
		D2=XDIST(XBS,XP)-S1	TRACK 92	
		D2=-D2	TRACK 93	
		IF(KHIT.GT.0)GOTO 105	TRACK 94	
		KHIT=KHIT+1	TRACK 95	
		MTARG=0	TRACK 96	
	105	KHIT=KHIT-1	TRACK 97	
		IH=ABS(H/CELSIZ)+.5	TRACK 98	
		IF(H.LT.0.)IH=-IH	TRACK 99	
		IV=ABS(V/CELSIZ)+.5	TRACK 100	
		IF(V.LT.0.)IV=-IV	TRACK 101	
C		PRINT CARD NUM 1	TRACK 102	
		IF(IIAPES.EQ.NO)GOTO 110	TRACK 103	
		WRITE (6,904)	TRACK 104	
		WRITE (6,901)HREF,VREF,IVIH,D1,D2,MSKRT,MTARG,MARMR,MVOL,	TRACK 105	
	1	KHIT,IH,IV,H,V	TRACK 106	
	110	IF(IWOT.EQ.IYES)WRITE(1,901)HREF,VREF,IVIH,D1,D2,MSKRT,MTARG,	TRACK 107	
	1	MARMR,MVOL,KHIT,IH,IV,H,V	TRACK 108	
C			TRACK 109	
C		PROCESS COMPONENT CARDS	TRACK 110	
C			TRACK 111	
		LMAX=L	TRACK 112	
		L=0	TRACK 113	
		TRAVEL=TR(1)	TRACK 114	
C			TRACK 115	
C	NIR	REGION IDENTIFICATION(VEHICLE COMPONENT)	TRACK 116	
C	SIN	LINE-OF-SIGHT DISTANCE	TRACK 117	
C	ANGLE	OBLIQUITY ANGLE	TRACK 118	
C	SN	NORMAL DISTANCE THRU REGION	TRACK 119	
C	NTYPE	TYPE OF SPACE AFTER NIR(NONE=0,END RAY=9)	TRACK 120	
C	SSPACE	LINE-OF-SIGHT DISTANCE THRU SPACE	TRACK 121	
C			TRACK 122	
	DO 200 KIK=1,LMAX,2		TRACK 123	
	JERQ=1		TRACK 124	
	L=L+1		TRACK 125	
	IF(L.GE.LMAX)RETURN		TRACK 126	
	CALL CALC		TRACK 127	
	IF(NIR.NE.0)GOTO 113	102	TRACK 128	

JERRO=2	TRACK129
IERR0=IERR0+1	TRACK130
113 IF(SSPACE.NE.0.)JCNT=JCNT+1	TRACK131
NIR1=NIR	TRACK132
SIN1=SIN	TRACK133
ANGLE1=ANGLE	TRACK134
SN1=SN	TRACK135
NTYPE1=NTYPE	TRACK136
SSPACE1=SSPACE	TRACK137
C SECOND HALF OF CARD	TRACK138
L=L+1	TRACK139
IF(L.LT.LMAX)GOTO 115	TRACK140
NIR=0	TRACK141
SIN=0.	TRACK142
ANGLE=0.	TRACK143
SN=0.	TRACK144
NTYPE=0	TRACK145
SSPACE=0.	TRACK146
GOTO 120	TRACK147
115 CALL CALC	TRACK148
IF(NIR.NE.0)GOTO 117	TRACK149
JERRO=2	TRACK150
IERR0=IERR0+1	TRACK151
117 IF(SSPACE.EQ.0.)GOTO 130	TRACK152
120 JCNT=JCNT+1	TRACK153
130 I1=0	TRACK154
I2=0	TRACK155
N=L-JCNT	TRACK156
C TRACK FLAG 501 IS TRACK EDGE 502 IS TRACK FACE	TRACK157
C 10 IN. NORMAL THICKNESS IS CUTOFF	TRACK158
C	TRACK159
IF(NIR1.NE.501)GOTO 140	TRACK160
IF(SN1.LT.10.)NIR1=502	TRACK161
140 IF(NIR.NE.501)GOTO 150	TRACK162
IF(SN.LT.10.)NIR=502	TRACK163
C	TRACK164
C PRINT COMPONENT CARD	TRACK165
C	TRACK166
150 IF(IWOT.EQ.IYES)WRITE(1,902)NIR1,SIN1,SN1,ANGLE1,NTYPE1,SSPACE1,	TRACK167
1 NIR,SIN,SN,ANGLE,NTYPE,SSPACE,IH,IV,I1,I2,N	TRACK168
IF(ITAPE8.EQ.IYES)WRITE(6,902)NIR1,SIN1,SN1,ANGLE1,NTYPE1,SSPACE1,	TRACK169
1 NIR,SIN,SN,ANGLE,NTYPE,SSPACE,IH,IV,I1,I2,N,ERROR(JERRO)	TRACK170
IF(ITAPE8.EQ.NO.AND.JERRO.EQ.2)WRITE(6,905)IH,IV,HREF,VREF	TRACK171
C	TRACK172
IF(L.GE.LMAX)RETURN	TRACK173
IF(NTYPE.EQ.9)RETURN	TRACK174
200 CONTINUE	TRACK175
RETURN	TRACK176
END	TRACK177
C	TRACK178
C	TRACK179
SUBROUTINE CALC	TRACK180
DIMENSION MASTER(30000),XP(3),TEMP(3),TEMP1(3),TEM(3),TEM1(3),	**** 23
1 XMID(3),IEMP(4),WN(3),WI(3),WA(3),XI(3),AUN(3),HF(3),	CALC 2
2 VF(3),Q(3),DELTA(3),ARSTP(3)	CALC 3
COMMON ASTER(30000)	CALC 4
COMMON/PAREM/XB(3),WB(3),IR	CALC 5
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	CALC 6
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	CALC 7
	CALC 8

1	LPDATA,LCIN,LRAT,LIO,LOCDA,IIS,I30,LBODY,NASC,KLOOP	CALC	9
	COMMON/ATRACK/D1,D2,KHIT,LMAX,TR(200),XBS(3),IRSTRT,IENC,	CALC	10
1	ITYPE(2),CA,CE,SA,SE	CALC	11
	COMMON/ALZ/NIR,SIN,ANGLE,NTYPE,SPACE,L,XS(3),WS(3),TRAVEL,	CALC	12
1	SN,VN,IVN	CALC	13
	COMMON/WALT/LTRC,NUTRR	CALC	14
	EQUIVALENCE(MASTER,ASTER)	CALC	15
	REAL(DEF)	CALC	16
		CALC	17
901	FORMAT(10,15)('THATS ALL FOLKS//')	CALC	18
902	FORMAT(10,17)('I TYPE IN CALC,5X,6H I TYPE=,15,4H NBO=,15/	CALC	19
1	100 RETURN TO TRACK//')	CALC	20
903	FORMAT(10,23)('HARS DID NOT FIND NORMAL')	CALC	21
904	FORMAT(10,50)('NORMAL/5H NIR=,110,5X,6H I TYPE=,110,5X,	CALC	22
1	6H NBO=,110/4H NBS=,3E20.10/4H WS=,3E20.10/4H XP=,3E20.8/	CALC	23
2	4H XN=,3E20.10/4H XI=,3E20.10/6H XNOS=,3E20.10)	CALC	24
905	FORMAT(10,25)('ERROR IN CALC A TRC HAS R1 = R2')	CALC	25
906	FORMAT(10,26)('ERROR IN CALC BAD LSURF FOR BOX OR RAW')	CALC	26
		CALC	27
	7 BOXES / RPP / BODY NUM / NEXT REGION /	CALC	28
		CALC	29
	CALL OPNKK(LSURF,NIR,NIR)	CALC	30
	IF(LN.O1.O1)GOTO 10	CALC	31
	WRITE (6,901)	CALC	32
	RETURN	CALC	33
		CALC	34
	TRAVEL LINE-OF-SIGHT DIST TO THIS REGION	CALC	35
	AS STARTING POINT (XS=XD)	CALC	36
	SIN LINE-OF-SIGHT DIST THRU THIS REGION	CALC	37
		CALC	38
10	SIN=TR(L+1)	CALC	39
	DO 20 I=1,3	CALC	40
	XI(I)=XS(I)+TRAVEL*WS(I)	CALC	41
20	CONTINUE	CALC	42
	TRAVEL=TRAVEL+SIN	CALC	43
	LSURF=LSURF-7	CALC	44
		CALC	45
	XNOS=1.	CALC	46
	IF(LSURF.LT.0)XNOS=-1.	CALC	47
	LUC=LBODY+3*(NBO-1)	CALC	48
	CALL UN2(LUC,ITYPE,LDATA)	CALC	49
	LSURF=1A5(LSURF)	CALC	50
	ITYPE=ITYPE+1	CALC	51
	IF(ITYPE.GE.1.AND.ITYPE.LE.12)GOTO 30	CALC	52
	WRITE (6,902)ITYPE,NBO	CALC	53
	RETURN	CALC	54
		CALC	55
	COMPUTE NORMAL DIST AND OBLIQUITY ANGLE	CALC	56
		CALC	57
	RPP BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	CALC	58
30	GOTO(50,100,150,200,200,300,350,400,450,500,550,600),ITYPE	CALC	59
		CALC	60
	CHECK FOR SPACE CODES IDENT = -1,1-9,11-19,21-29,.....,91-99	CALC	61
		CALC	62
40	CALL OPENK(L+1,DUM,DUM,NEXREG)	CALC	63
	ISPOT=LIRFO+NEXREG-1	CALC	64
	CALL UN2(ISPOT,DUM,IDENT)	CALC	65
	ISPOT=LIRFO+NIR-1	CALC	66
	CALL UN2(ISPOT,NIR,DUM)	CALC	67
	IDENT=IDENT-1	CALC	68

C	CHECK FOR SPACE CODES IDENT = -1,1-9,11-19,21-29,.....	CALC 69
	IF(IDENT-(IDENT/10)*10.NE.0)GOTO 41	CALC 70
	NTYPE=0	CALC 71
	SSPACE=0.	CALC 72
	RETURN	CALC 73
41	L=L+1	CALC 74
	IF(L+1.LT.LMAX)GOTO 42	CALC 75
	IDENT=9	CALC 76
	SSPACE=1.0E-4	CALC 77
	NTYPE=IDENT	CALC 78
	RETURN	CALC 79
42	NTYPE=IDENT	CALC 80
	SSPACE=TR(L+1)	CALC 81
	TRAVEL=TRAVEL+SSPACE	CALC 82
	RETURN	CALC 83
C		CALC 84
C	RPP	CALC 85
C		CALC 86
50	IF(LSURF-2)52,53,54	CALC 87
52	XNOS=-XNOS	CALC 88
53	I=1	CALC 89
	GOTO 60	CALC 90
54	IF(LSURF-4)55,56,57	CALC 91
55	XNOS=-XNOS	CALC 92
56	I=3	CALC 93
	GOTO 60	CALC 94
57	IF(LSURF.GE.6)GOTO 59	CALC 95
	XNOS=-XNOS	CALC 96
59	I=5	CALC 97
60	LKK=LBASF+2*I+1	CALC 98
	LV1=MASTER(LKK)	CALC 99
	LKK=LKK+2	CALC 100
	LV2=MASTER(LKK)	CALC 101
	DO 62 J=1,3	CALC 102
	M=J-1	CALC 103
	IJK=M+LV1	CALC 104
	TEMP(J)=ASTER(IJK)	CALC 105
	IJK=M+LV2	CALC 106
	TEMP1(J)=ASTER(IJK)	CALC 107
62	CONTINUE	CALC 108
	CALL DCOSP(TEMP,TEMP1,WB)	CALC 109
	DO 63 J=1,3	CALC 110
	WB(J)=XNOS*WB(J)	CALC 111
63	CONTINUE	CALC 112
	GOTO 1000	CALC 113
C		CALC 114
C	BOX	CALC 115
C		CALC 116
100	CONTINUE	CALC 117
	KCOM=LSURF-(LSURF/2)*2	CALC 118
	IF(KCOM.EQ.0)XNOS=-XNOS	CALC 119
	IF(LSURF-3)104,103,105	CALC 120
103	I=1	CALC 121
	GOTO 110	CALC 122
104	I=2	CALC 123
	GOTO 110	CALC 124
105	IF(LSURF.LT.5)GOTO 103	CALC 125
	I=3	CALC 126
110	CALL UN2(LDATA,TEMP(4),TEMP(1))	CALC 127
	LDATA=LDATA+1	CALC 128

CALL UN2(LDATA,TEMP(2),TEMP(3))	CALC 129
DO 115 J=1,3	CALC 130
LH=TEMP(1)	CALC 131
LV=TEMP(4)	CALC 132
M=J-1	CALC 133
IJK=LH+M	CALC 134
IJK1=LV+M	CALC 135
TEMP(J)=ASTER(IJK)+ASTER(IJK1)	CALC 136
MK=J-1+TEMP(4)	CALC 137
TEMP1(J)=ASTER(MK)	CALC 138
115 CONTINUE	CALC 139
CALL DCOSP(TEMP1,TEMP,WB)	CALC 140
DO 120 J=1,3	CALC 141
WB(J)=XNOS*WB(J)	CALC 142
120 CONTINUE	CALC 143
GOTO 1000	CALC 144
C	CALC 145
C SPH	CALC 146
C	CALC 147
150 CALL UN2(LDATA,LV,DUM)	CALC 148
DO 160 I=1,3	CALC 149
M=I-1+LV	CALC 150
TEM(I)=ASTER(M)	CALC 151
160 CONTINUE	CALC 152
CALL DCOSP(XI,TEM,WB)	CALC 153
DO 170 I=1,3	CALC 154
WB(I)=XNOS*WB(I)	CALC 155
170 CONTINUE	CALC 156
GOTO 1000	CALC 157
C	CALC 158
C RCC	CALC 159
C	CALC 160
200 IF(LSURF-2)202,201,210	CALC 161
201 XNOS=-XNOS	CALC 162
202 CALL UN2(LDATA,LV1,LV2)	CALC 163
DO 203 I=1,3	CALC 164
M=I-1	CALC 165
IJK1=M+LV1	CALC 166
IJK2=M+LV2	CALC 167
TEM(I)=ASTER(IJK1)	CALC 168
TEM1(I)=ASTER(IJK1)+ASTER(IJK2)	CALC 169
203 CONTINUE	CALC 170
CALL DCOSP(TEM,TEM1,WB)	CALC 171
DO 204 I=1,3	CALC 172
WB(I)=XNOS*WB(I)	CALC 173
204 CONTINUE	CALC 174
GOTO 1000	CALC 175
C	CALC 176
C DIR COS FOR NORMAL TO SURFACE ONE OR TWO	CALC 177
C NOW HAVE TO GET FROM A POINT TO THE HEIGHT VECTOR	CALC 178
C	CALC 179
210 CALL UN2(LDATA,LV,LH)	CALC 180
LRL=MASTER(LDATA+1)	CALC 181
DO 211 J=1,3	CALC 182
M=J-1	CALC 183
IJK=LV+M	CALC 184
TEM(J)=ASTER(IJK)	CALC 185
IJK1=LH+M	CALC 186
TEM1(J)=ASTER(IJK)+ASTER(IJK1)	CALC 187
211 CONTINUE	CALC 188

CALL DCOSP(TEM,XI,WN)	CALC 189
CALL DCOSP(TEM,TEM1,WI)	CALC 190
SUM=0.	CALC 191
DO 212 J=1,3	CALC 192
SUM=SUM+WN(J)*WI(J)	CALC 193
212 CONTINUE	CALC 194
DO 214 J=1,3	CALC 195
XP(J)=SUM*XDIST(TEM,XI)	CALC 196
XP(J)=XP(J)*WI(J)+TEM(J)	CALC 197
214 CONTINUE	CALC 198
IF(IITYPE.EQ.5)GOTO 250	CALC 199
CALL DCCSP(XI,XP,WB)	CALC 200
DO 220 J=1,3	CALC 201
WB(J)=XNOS*WB(J)	CALC 202
220 CONTINUE	CALC 203
GOTO 1000	CALC 204
C	CALC 205
C REC	CALC 206
C	CALC 207
C FOR SURFACE 1 AND 2 NORMAL IS SAME AS RCC	CALC 208
C FOR SURFACE 3 JUMP OUT WHEN XP(I)=POINT ON HEIGHT VECTOR	CALC 209
C	CALC 210
250 LDAIA=LDAIA+1	CALC 211
CALL UN2(LDAIA,LR1,LR2)	CALC 212
DO 255 J=1,3	CALC 213
M=J-1	CALC 214
IJK1=M+LR1	CALC 215
TEMP(J)=ASTER(IJK1)+XP(J)	CALC 216
IJK2=M+LR2	CALC 217
TEMP1(J)=ASTER(IJK2)+XP(J)	CALC 218
255 CONTINUE	CALC 219
A1=XDIST(XP,TEMP)	CALC 220
A2=XDIST(XP,TEMP1)	CALC 221
IF(A1.GE.A2)GOTO 260	CALC 222
A1=A2	CALC 223
A3=A1	CALC 224
A2=A3	CALC 225
TEMP(1)=TEMP1(1)	CALC 226
TEMP(2)=TEMP1(2)	CALC 227
TEMP(3)=TEMP1(3)	CALC 228
260 C=SQRT(A1*A1-A2*A2)	CALC 229
CALL DCOSP(XP,TEMP,WN)	CALC 230
DO 265 J=1,3	CALC 231
TEM(J)=XP(J)+C*WN(J)	CALC 232
TEM1(J)=XP(J)-C*WN(J)	CALC 233
265 CONTINUE	CALC 234
CALL DCOSP(TEM,XI,WN)	CALC 235
DO 270 J=1,3	CALC 236
TEM(J)=2.*A1*WN(J)+TEM(J)	CALC 237
270 CONTINUE	CALC 238
CALL DCOSP(TEM,TEM1,WB)	CALC 239
DO 275 J=1,3	CALC 240
WB(J)=XNOS*WB(J)	CALC 241
275 CONTINUE	CALC 242
GOTO 1000	CALC 243
C	CALC 244
C TRC	CALC 245
C	CALC 246
300 IF(LSURF.LE.2)GO TO 320	CALC 247
CALL UN2(LDAIA,LV,LH)	CALC 248

LDATA=LDATA+1	CALC 247
CALL UNZ(LDATA,LR1,LR2)	CALC 248
DIF=ASTER(LR1)-ASTER(LR2)	CALC 249
IF(DIF)302,301,303	CALC 250
301 WRITE (6,905)	CALC 251
STOP	CALC 252
302 TEMP(1)=LR1	CALC 253
LR1=LR2	CALC 254
LR2=TEMP(1)	CALC 255
DIF=ABS(DIF)	CALC 256
303 FACTR=ASTER(LR1)/DIF	CALC 257
DO 304 J=1,3	CALC 258
M=J-1	CALC 259
IJK=M+LV	CALC 260
IJK1=M+LH	CALC 261
TEMP1(J)=ASTER(IJK)	CALC 262
TEMP(J)=ASTER(IJK)+FACTR*ASTER(IJK1)	CALC 263
304 CONTINUE	CALC 264
IDIS=XDIS(XI,TEMP)	CALC 265
QDIS=XDIS(TEMP1,TEMP)	CALC 266
CALL DCOSP(TEMP,XI,WN)	CALC 267
CALL DCOSP(TEMP,TEMP1,WA)	CALC 268
SUM=0.	CALC 269
DO 310 J=1,3	CALC 270
SUM=WN(J)*WA(J)+SUM	CALC 271
310 CONTINUE	CALC 272
QSUM=IDIS/SUM	CALC 273
QPLS=QDIS-QSUM	CALC 274
DO 311 J=1,3	CALC 275
TEMP(J)=-QPLS*WA(J)+TEMP1(J)	CALC 276
311 CONTINUE	CALC 277
CALL DCOSP(XI,TEMP,WB)	CALC 278
DO 312 J=1,3	CALC 279
WB(J)=XNOS*WB(J)	CALC 280
312 CONTINUE	CALC 281
GOTO 1000	CALC 282
C	CALC 283
320 IF(LSURF.EQ.2)XNOS=-XNOS	CALC 284
CALL UNZ(LDATA,LV,LH)	CALC 285
DO 321 J=1,3	CALC 286
M=J-1	CALC 287
IJK=M+LV	CALC 288
TEMP(J)=ASTER(IJK)	CALC 289
IJK1=M+LH	CALC 290
TEMP1(J)=ASTER(IJK)+ASTER(IJK1)	CALC 291
321 CONTINUE	CALC 292
CALL DCOSP(TEMP,TEMP1,WB)	CALC 293
DO 322 J=1,3	CALC 294
WB(J)=XNOS*WB(J)	CALC 295
322 CONTINUE	CALC 296
GOTO 1000	CALC 297
C	CALC 298
C	CALC 299
C	CALC 300
350 CALL UNZ(LDATA,LR1,LR2)	CALC 301
LS=MASTER(LDATA+1)	CALC 302
DO 352 J=1,3	CALC 303
M=J-1	CALC 304
IJK1=M+LR1	CALC 305
IJK2=M+LR2	CALC 306

```

      TEM(J)=ASTER(IJK1)
      TEM1(J)=ASTER(IJK2)
352 CONTINUE
      A=ASTER(LS)
      CALL DCOSP(TEM,XI,WN)
      DO 353 J=1,3
      TEM(J)=A*WN(J)+TEM(J)
353 CONTINUE
      CALL DCOSP(TEM,TEM1,WB)
      DO 354 J=1,3
      WB(J)=XNOS*WB(J)
354 CONTINUE
      GOTO 1000
C
C      RAW
C
C      THIS WILL SHAPE THE BOX FOR LSURF=1,3,5,6
C      JUMPS TO 100 TO INDICATE BOX PORTION
C
400 IF(LSURF.EQ.2)GOTO 401
      IF(LSURF.NE.4)GOTO 100
      WRITE (6,906)
      STOP
401 CALL UN2(LDATA,LV,LV1)
      LDATA=LDATA+1
      CALL UN2(LDATA,LV2,LV3)
      DO 410 J=1,3
      M=J-1
      IJK1=M+LV1
      IJK2=M+LV2
      TEMP(J)=ASTER(IJK1)
      XMID(J)=ASTER(IJK1)-ASTER(IJK2)
      IJK3=M+LV3
      TEM(J)=ASTER(IJK3)
410 CONTINUE
      I=1
      J=2
      K=3
      LK=0
      DO 411 KK=1,3
      TEM1(I)=XMID(J)*TEM(K)-XMID(K)*TEM(J)
      LK=I
      I=J
      J=K
      K=LK
411 CONTINUE
      SUM=0.
      DO 412 J=1,3
      SUM=TEM1(J)*TEMP(J)+SUM
412 CONTINUE
      SUM=-SUM/ABS(SUM)
      TLK=TEM1(1)**2+TEM1(2)**2+TEM1(3)**2
      TLK=SQRT(TLK)
      DO 420 J=1,3
      WB(J)=XNOS*SUM*TEM1(J)/TLK
420 CONTINUE
      GOTO 1000

```

ARB

109

CALC 309
 CALC 310
 CALC 311
 CALC 312
 CALC 313
 CALC 314
 CALC 315
 CALC 316
 CALC 317
 CALC 318
 CALC 319
 CALC 320
 CALC 321
 CALC 322
 CALC 323
 CALC 324
 CALC 325
 CALC 326
 CALC 327
 CALC 328
 CALC 329
 CALC 330
 CALC 331
 CALC 332
 CALC 333
 CALC 334
 CALC 335
 CALC 336
 CALC 337
 CALC 338
 CALC 339
 CALC 340
 CALC 341
 CALC 342
 CALC 343
 CALC 344
 CALC 345
 CALC 346
 CALC 347
 CALC 348
 CALC 349
 CALC 350
 CALC 351
 CALC 352
 CALC 353
 CALC 354
 CALC 355
 CALC 356
 CALC 357
 CALC 358
 CALC 359
 CALC 360
 CALC 361
 CALC 362
 CALC 363
 CALC 364
 CALC 365
 CALC 366
 CALC 367
 CALC 368

T3=DOT(TEMP,AUN)/TAU	CALC 429
T4=DOT(TEMP,C)	CALC 430
GAMMA=ODN/HDN	CALC 431
EM=GAMMA*R4+(1.0-GAMMA)*R2	CALC 432
T5=HDA/HDN	CALC 433
T6=HDK/HDN	CALC 434
DO 510 I=1,3	CALC 435
WB(I)=XNOS*(T3*(AUN(I)-T5*NF(I))+	CALC 436
1 T4*(O(I)-T6*NF(I))-EM*(R4-R2)*NF(I)/HDN)	CALC 437
510 CONTINUE	CALC 438
CALL UNIT(WB)	CALC 439
GOTO 1000	CALC 440
C	CALC 441
520 IF(LSURF.EQ.2)XNOS=-XNOS	CALC 442
CALL UN2(LDATA,LV,LH)	CALC 443
LDATA=LDATA+1	CALC 444
CALL UN2(LDATA,LN,DUM)	CALC 445
DO 521 I=1,3	CALC 446
J=LN+I-1	CALC 447
WB(I)=XNOS*ASTER(J)	CALC 448
521 CONTINUE	CALC 449
GOTO 1000	CALC 450
C	CALC 451
C FOR	CALC 452
C	CALC 453
550 CALL UN2(LDATA,LV,LN)	CALC 454
LDATA=LDATA+1	CALC 455
CALL UN2(LDATA,LR1,DUM)	CALC 456
DO 551 I=1,3	CALC 457
J=I-1	CALC 458
IJK=LV+J	CALC 459
TEMP(I)=XI(I)-ASTER(IJK)	CALC 460
IJK=LN+J	CALC 461
TEMP1(I)=ASTER(IJK)	CALC 462
551 CONTINUE	CALC 463
R1=ASTER(LR1)	CALC 464
CALL CROSS(TEM,TEMP1,TEMP)	CALC 465
CALL CROSS(TEMP1,TEM,TEMP1)	CALC 466
CALL UNIT(TEMP1)	CALC 467
DO 552 I=1,3	CALC 468
J=I-1	CALC 469
IJK=LV+J	CALC 470
TEM(I)=ASTER(IJK)	CALC 471
TEMP1(I)=TEM(I)+R1*TEMP1(I)	CALC 472
552 CONTINUE	CALC 473
CALL DCOSP(TEMP1,XI,WB)	CALC 474
DO 553 I=1,3	CALC 475
WB(I)=XNOS*WB(I)	CALC 476
553 CONTINUE	CALC 477
GOTO 1000	CALC 478
C	CALC 479
C ARS	CALC 480
C	CALC 481
600 NE=4	CALC 482
IWH=MASTER(LDATA)	CALC 483
INOW=IWH+8	CALC 484
IEND=IWH+8+20*NE	CALC 485
DO 610 I=1,3	CALC 486
IJK=IWH+I+4	CALC 487
ARSTP(I)=ASTER(IJK)	CALC 488

610	CONTINUE	CALC 489
	DTRAV=XDIST(ARSTP,XI)	CALC 490
620	IF(ABS(DTRAV-ASTER(INOW)).GT.1.0E-07)GOTO 640	CALC 491
	DO 630 I=1,3	CALC 492
	IJK=INOW+I	CALC 493
	WB(I)=ASTER(IJK)	CALC 494
630	CONTINUE	CALC 495
	CALL UNIT(WB)	CALC 496
	GOTO 1000	CALC 497
C		CALC 498
640	INOW=INOW+NE	CALC 499
	IF(IEND.GT.INOW)GOTO 620	CALC 500
	WRITE (6,903)	CALC 501
	STOP	CALC 502
C		CALC 503
C	COMPUTE OBLIQUITY ANGLE	CALC 504
C	COMPUT NORMAL DIST (SN)	CALC 505
C		CALC 506
1000	DO 1001 J=1,3	CALC 507
	XB(J)=XI(J)+WS(J)*1.0E-3	CALC 508
1001	CONTINUE	CALC 509
	ANGLE=0.	CALC 510
	DO 1002 J=1,3	CALC 511
	ANGLE=ANGLE+WB(J)*WS(J)	CALC 512
1002	CONTINUE	CALC 513
	IF(ABS(ANGLE).LE.1.)GOTO 1010	CALC 514
	ANGLE=0.	CALC 515
	SN=0.	CALC 516
	WRITE (6,904)NIR,IITYPE,NBO,LSURF,WB,WS,XP,XB,XI,XNOS	CALC 517
	IR=NIR	CALC 518
	GOTO 40	CALC 519
C		CALC 520
1010	ANGLE=ATAN2(SQRT(1.-ANGLE*ANGLE),ANGLE)*180./3.141592654	CALC 521
	IF(ANGLE.LE.90.)GOTO 1020	CALC 522
	DO 1011 J=1,3	CALC 523
	WB(J)=-WB(J)	CALC 524
1011	CONTINUE	CALC 525
	GOTO 1000	CALC 526
C		CALC 527
1020	NASC=-2	CALC 528
	IR=NIR	CALC 529
	CALL G1(S1,IRPRIM,XP)	CALC 530
	SN=S1	CALC 531
	GOTO 40	CALC 532
	END	CALC 533
C		CALC 534
C		CALC 535
C		CALC 536
C		CALC 537
C		CALC 538
	SUBROUTINE TESTG	**** 24
C		TESTG 2
C	TESTG OPTIONS	TESTG 3
C		TESTG 4
C	NRAYS 0 0 TRACE A RAY BETWEEN TWO GIVEN POINTS	TESTG 5
C	XBS TO XBF	TESTG 6
C		TESTG 7
	DIMENSION XP(3),XBF(3)	TESTG 8
	COMMON/PAREM/XB(3),WB(3),IR	TESTG 9
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	TESTG 10

```

COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
COMMON/WALT/LIRFO,NGIERR

```

```

C
901 FORMAT(2I10)
902 FORMAT(1H0,22HNUMBER OF SPECIAL RAYS,I5)
903 FORMAT(3E15.7,3I15)
904 FORMAT(1H0,5HSTART,5X,4H XB=,3E15.7,8H IRSTRT=,I5/
1  4H END,7X,4HXB=,3E15.7,8H IRFIN=,I5)
905 FORMAT(1H0,3HXB=,3E15.7,5X,6HRANGE=,E15.7)
906 FORMAT(1H0,8X,2HIR,4X,6HIRPRIM,12X,2HS1,13X,2HXP,13X,2HYP,
1  13X,2HZP,12X,4HDIST)
907 FORMAT(2I10,5X,5E15.7)
908 FORMAT(1H0,21HTROUBLE IN REGION IR=,I10)

C
READ (5,901)NRAYS,NGIERR
WRITE (6,902)NRAYS
IF(NGIERR.LE.0)NGIERR=25

C
DO 50 IRAY=1,NRAYS
READ (5,903)XB,IRSTRT
READ (5,903)XBF,IRFIN
WRITE (6,904)XB,IRSTRT,XBF,IRFIN
RANGE=XDIST(XB,XBF)
CALL DCOSP(XB,XBF,WB)
WRITE (6,905)WB,RANGE
IR=IRSTRT
NASC=-1
WRITE (6,906)

C
10 CALL G1(S1,IRPRIM,XP)
IF(IERR.GE.NGIERR)GOTO 60
WRITE (6,907)IR,IRPRIM,S1,XP,DIST
IF(DIST.GE.RANGE)GOTO 30
IF(IRPRIM.LE.0)GOTO 20
IR=IRPRIM
GOTO 10

C
20 WRITE (6,908)IR
GOTO 50
30 IF(IR.NE.IRFIN)GOTO 20
50 CONTINUE
60 IERR=0
RETURN
END.

C
C
C
C
SUBROUTINE VOLUM
DIMENSION VASTER(1000),WAB(3),WTB(3),WOB(3),DSP(3),
1  XV(3),XT(3),XA(3),XO(3),XP(3),XTEMP(3)
COMMON ASTER(30000)
COMMON/PAREM/XB(3),WB(3),IR
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
1  LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
COMMON/WALT/LIRFO,NGIERR

C
901 FORMAT(3E20.8)

```

```

TESTG 11
TESTG 12
TESTG 13
TESTG 14
TESTG 15
TESTG 16
TESTG 17
TESTG 18
TESTG 19
TESTG 20
TESTG 21
TESTG 22
TESTG 23
TESTG 24
TESTG 25
TESTG 26
TESTG 27
TESTG 28
TESTG 29
TESTG 30
TESTG 31
TESTG 32
TESTG 33
TESTG 34
TESTG 35
TESTG 36
TESTG 37
TESTG 38
TESTG 39
TESTG 40
TESTG 41
TESTG 42
TESTG 43
TESTG 44
TESTG 45
TESTG 46
TESTG 47
TESTG 48
TESTG 49
TESTG 50
TESTG 51
TESTG 52
TESTG 53
TESTG 54
TESTG 55
TESTG 56
TESTG 57
TESTG 58
TESTG 59
**** 25
VOLUM 2
VOLUM 3
VOLUM 4
VOLUM 5
VOLUM 6
VOLUM 7
VOLUM 8
VOLUM 9
VOLUM 10
VOLUM 11

```

902	FORMAT(2E20.8)	VOLUM 12
903	FORMAT(1H0,10X,6HVERTEX,14X,6HTOP.PT,14X,6HBOT.PT,14X,7HSIDE.PT)	VOLUM 13
904	FORMAT(4E20.8)	VOLUM 14
905	FORMAT(1H0, 8X,12HDELTA ON TOP,E20.8,10X,10HSIDE DELTA,E20.8)	VOLUM 15
906	FORMAT(2110)	VOLUM 16
908	FORMAT(1H0, 2X,1RHSTARTING REGION IS,15)	VOLUM 17
909	FORMAT(1H0,16HVASTER OVERWRITE,5X,6HNRMAX=,15)	VOLUM 18
910	FORMAT(110,E20.8)	VOLUM 19
911	FORMAT(1H0,8HBAD CARD/110,E20.8,14H NOT PROCESSED)	VOLUM 20
912	FORMAT(110,E20.8,5X,E20.8,5X,E9.2)	VOLUM 21
913	FORMAT(1H0,5HSUMV=,5X,E20.8)	VOLUM 22
C		VOLUM 23
	READ (5,906)IR,NGIERR	VOLUM 24
	IF(NGIERR.LE.0)NGIERR=25	VOLUM 25
	READ (5,901)(XV(I),I=1,3)	VOLUM 26
	READ (5,901)(XT(I),I=1,3)	VOLUM 27
	READ (5,901)(XO(I),I=1,3)	VOLUM 28
	READ (5,901)(XA(I),I=1,3)	VOLUM 29
	READ (5,902)DOD,DT	VOLUM 30
	WRITE (6,903)	VOLUM 31
	WRITE (6,904)(XV(J),XT(J),XO(J),XA(J),J=1,3)	VOLUM 32
	WRITE (6,905)DOD,DT	VOLUM 33
	WRITE (6,908)IR	VOLUM 34
	IF(NRMAX.GT.2000)WRITE (6,909)NRMAX	VOLUM 35
	CALL DCOSP(XV,XT,WTB)	VOLUM 36
	CALL DCOSP(XV,XO,WOB)	VOLUM 37
	CALL DCOSP(XV,XA,WAB)	VOLUM 38
	XVDIS=XDIST(XV,XA)	VOLUM 39
	TESTUN=0.	VOLUM 40
	TESTOV=0.	VOLUM 41
	XTMP(1)=0.	VOLUM 42
	DO 10 I=1,NRMAX	VOLUM 43
	VASTER(I)=0.	VOLUM 44
10	CONTINUE	VOLUM 45
	JIR=IR	VOLUM 46
	IRJ=IR	VOLUM 47
	N2=XDIST(XV,XO)/DOD+1.	VOLUM 48
	N1=XDIST(XV,XT)/DT+1.	VOLUM 49
C		VOLUM 50
	DO 300 J=1,N2	VOLUM 51
	DO 100 I=1,3	VOLUM 52
	DSP(I)=WTB(I)*DT	VOLUM 53
	XB(I)=XV(I)	VOLUM 54
	WB(I)=WAB(I)	VOLUM 55
100	CONTINUE	VOLUM 56
	S1=0.	VOLUM 57
	IR=JIR	VOLUM 58
	DO 200 I=1,N1	VOLUM 59
	NASC=-1	VOLUM 60
110	CALL G1(S1,IRPRIM,XP)	VOLUM 61
	IF(IERR.GE.NGIERR)GOTO 400	VOLUM 62
	VASTER(IR)=VASTER(IR)+S1	VOLUM 63
	IF(DIST.GE.XVDIS)GOTO 115	VOLUM 64
	IF(IRPRIM.LE.0)GOTO 120	VOLUM 65
	IR=IRPRIM	VOLUM 66
	GOTO 110	VOLUM 67
115	VASTER(IR)=VASTER(IR)-(DIST-XVDIS)	VOLUM 68
120	XTMP(1)=WB(1)	VOLUM 69
	XTMP(2)=WB(2)	VOLUM 70
	XTMP(3)=WB(3)	VOLUM 71

```

IR=JIR
TESTDN=TESTDN-DT
IF(TESTDN.GT.0.)GOTO 180
WB(1)=WTB(1)
WB(2)=WTB(2)
WB(3)=WTB(3)
NASC=-1
CALL G1(S1,IRPRIM,XP)
IF(IERR.GE.NG1ERR)GOTO 400
IF(S1-DT)130,160,170
130 IR=IRPRIM
JIR=IR
CALL G1(S1,IRPRIM,XP)
IF(IERR.GE.NG1ERR)GOTO 400
IF(DIST-DT)140,160,170
140 IF(IRPRIM)150,210,130
150 STOP
160 IR=IRPRIM
JIR=IR
170 TESTDN=S1
180 DO 190 JI=1,3
WB(JI)=XTEMP(JI)
XB(JI)=XB(JI)+DSP(JI)
190 CONTINUE
200 CONTINUE
C
C ONE PLANE DONE - MOVE IN FOR NEXT PLANE IN LINE
C
210 NASC=-1
DO 220 I=1,3
WB(I)=WOB(I)
XB(I)=XV(I)
220 CONTINUE
JIR=IRJ
IR=JIR
TESTOV=0.
TESTOV=TESTOV-DOD
IF(TESTOV)230,230,280
230 CALL G1(S1,IRPRIM,XP)
IF(IERR.GE.NG1ERR)GOTO 400
IF(S1-DOD)240,260,270
240 IR=IRPRIM
IRJ=IR
CALL G1(S1,IRPRIM,XP)
IF(IERR.GE.NG1ERR)GOTO 400
IF(DIST-DOD)250,260,270
250 IF(IRPRIM)150,400,230
260 IR=IRPRIM
IRJ=IR
270 TESTOV=S1
280 DO 290 I=1,3
XA(I)=XA(I)+WOB(I)*DOD
XV(I)=XV(I)+WOB(I)*DOD
XT(I)=XT(I)+WOB(I)*DOD
290 CONTINUE
JIR=IR
300 CONTINUE
C
C VOLUMES COMPUTED
C

```

```

VOLUM 72
VOLUM 73
VOLUM 74
VOLUM 75
VOLUM 76
VOLUM 77
VOLUM 78
VOLUM 79
VOLUM 80
VOLUM 81
VOLUM 82
VOLUM 83
VOLUM 84
VOLUM 85
VOLUM 86
VOLUM 87
VOLUM 88
VOLUM 89
VOLUM 90
VOLUM 91
VOLUM 92
VOLUM 93
VOLUM 94
VOLUM 95
VOLUM 96
VOLUM 97
VOLUM 98
VOLUM 99
VOLUM100
VOLUM101
VOLUM102
VOLUM103
VOLUM104
VOLUM105
VOLUM106
VOLUM107
VOLUM108
VOLUM109
VOLUM110
VOLUM111
VOLUM112
VOLUM113
VOLUM114
VOLUM115
VOLUM116
VOLUM117
VOLUM118
VOLUM119
VOLUM120
VOLUM121
VOLUM122
VOLUM123
VOLUM124
VOLUM125
VOLUM126
VOLUM127
VOLUM128
VOLUM129
VOLUM130
VOLUM131

```

400 READ (5,910)IR1,VR	VOLUM132
IF(IERR.GE.NG1ERR)GOTO 500	VOLUM133
IF(IR1.LE.0)IR1=NRMAX+1	VOLUM134
SUMV=0.	VOLUM135
C	VOLUM136
DO 450 I=1,NRMAX	VOLUM137
VASTLR(I)=VASTER(I)*DOD*DT	VOLUM138
IF(I-IR1)410,430,420	VOLUM139
410 WRITE (6,910)I,VASTER(I)	VOLUM140
GOTO 440	VOLUM141
420 WRITE (6,911)IR1,VR	VOLUM142
READ (5,910)IR1,VR	VOLUM143
GOTO 410	VOLUM144
C	VOLUM145
VOLUME REPLACEMENT	VOLUM146
430 XPERC=100.*(VASTER(I)/VR-1.)	VOLUM147
WRITE (6,912)I,VASTER(I),VR,XPERC	VOLUM148
VASTLR(I)=VR	VOLUM149
READ (5,910)IR1,VR	VOLUM150
440 SUMV=SUMV+VASTER(I)	VOLUM151
450 CONTINUE	VOLUM152
WRITE (6,913)SUMV	VOLUM153
500 ILKK=L	VOLUM154
RETURN	VOLUM155
END	VOLUM156
C	VOLUM157
C	VOLUM158
C	VOLUM159
C	**** 26
SUBROUTINE AREA	AREA 2
DIMENSION XP(3),WP(3),XBS(3),CONVRT(4,4),TYPEUN(4)	AREA 3
COMMON ASTER(3000)	AREA 4
COMMON/PAREN/XB(3),WB(3),IR	AREA 5
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	AREA 6
COMMON/UNCLEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	AREA 7
1 LDATA,LRIN,LROI,LIO,LJCCA,I15,I30,LBODY,NASC,KLOOP	AREA 8
COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),	AREA 9
1 TRAVEL,SN,V,H,IVIH	AREA 10
COMMON/WALT/LIRFO,NG1ERR	AREA 11
COMMON/CELL/CELSIZ	AREA 12
COMMON/ENGCOM/LLGEOM	AREA 13
C	AREA 14
901 FORMAT(7I10,6X,2A2)	AREA 15
902 FORMAT(6E12.8)	AREA 16
908 FORMAT(1H0,22HMEMORY OVERLAP IN AREA,5X,7HLEGEOM=,I6,	AREA 17
1 5X,6HLEGEOM=,I6,5X,6HLEGEOM=,I6)	AREA 18
909 FORMAT(1H0,13HERROR IN AREA,5X,9HIC0DE = '0)	AREA 19
910 FORMAT(1H0,8HAZIMUTH=,F10.3,5X,10HELEVATION=,F10.3)	AREA 20
911 FORMAT(1H0,12HCELL SIZE IS,F4.1,1X,1HX,F4.1,1X,A2,1H.,10X,	AREA 21
1 12HAREAS IN SQ.,1X,A2,1H.)	AREA 22
912 FORMAT(1H0,5HIC0DE,19X,4HAREA/)	AREA 23
913 FORMAT(15,15X,F12.5)	AREA 24
914 FORMAT(1H0,15HPRESENTED AREA=,F12.5)	AREA 25
915 FORMAT(1H0,18HNUMBER OF CELLS IS,I5,10X,	AREA 26
1 22HNUMBER OF CELLS HIT IS,I5)	AREA 27
C	AREA 28
C	AREA 29
C	AREA 30
C	AREA 31
DATA HHIN,HHFI,HHCM,HHMB,HHBB/2HIN,2HFT,2HCM,2HM ,2H /	AREA 32
TYPFUN(1)=HHIN	

TYPEUN(2)=HHFT	AREA 33
TYPEUN(3)=HHCM	AREA 34
TYPEUN(4)=HHMB	AREA 35
CONVRT(1,1)=1.	AREA 36
CONVRT(1,2)=.006944444444444444	AREA 37
CONVRT(1,3)=6.451625806	AREA 38
CONVRT(1,4)=.0006451625806	AREA 39
CONVRT(2,1)=144.	AREA 40
CONVRT(2,2)=1.	AREA 41
CONVRT(2,3)=929.0341161	AREA 42
CONVRT(2,4)=.09290341161	AREA 43
CONVRT(3,1)=.15499969	AREA 44
CONVRT(3,2)=.001076386736	AREA 45
CONVRT(3,3)=1.	AREA 46
CONVRT(3,4)=.0001	AREA 47
CONVRT(4,1)=1549.9969	AREA 48
CONVRT(4,2)=10.7636736	AREA 49
CONVRT(4,3)=10000.	AREA 50
CONVRT(4,4)=1.	AREA 51
BLANK=HHBB	AREA 52
C	AREA 53
LAREA=LIRFO-1000	AREA 54
IF(LAREA.GE.LEGEOM)GOTO 10	AREA 55
WRITE (6,908)LEGEOM,LAREA,LIRFO	AREA 56
STOP	AREA 57
10 LAREA1=LIRFO-1	AREA 58
DO 20 L=LAREA,LAREA1	AREA 59
ASTER(L)=0.	AREA 60
20 CONTINUE	AREA 61
C	AREA 62
READ (5,901)NX,NY,IRSTRT ,IENC,NGIERR,NSTART,NEND,CELLUN,AREAUN	AREA 63
READ (5,902)A,E,ENGTH,ZSHIFT,GROUND	AREA 64
READ (5,902)XSHIFT,YSHIFT,CELSIZ	AREA 65
IF(IRSTRT .LE.0)IRSTRT=1	AREA 66
IF(CELSIZ .LE.0.)CELSIZ=4.	AREA 67
IF(NSTART.LE.0)NSTART=1	AREA 68
IF(NGIERR.LE.0)NGIERR=25	AREA 69
IF(AREAUN.EQ.BLANK)AREAUN=HHIN	AREA 70
IF(CELLUN.EQ.BLANK)CELLUN=HHIN	AREA 71
DO 30 I=1,4	AREA 72
IF(CELLUN.EQ.TYPEUN(I))GOTO 40	AREA 73
30 CONTINUE	AREA 74
40 DO 50 J=1,4	AREA 75
IF(AREAUN.EQ.TYPEUN(J))GOTO 60	AREA 76
50 CONTINUE	AREA 77
60 AREA=CELSIZ *CELSIZ *CONVRT(I,J)	AREA 78
C	AREA 79
RADIAN=.017453292519943	AREA 80
AR=A*RADIAN	AREA 81
ER=E*RADIAN	AREA 82
SA=SIN(AR)	AREA 83
CA=COS(AR)	AREA 84
SE=SIN(ER)	AREA 85
CE=COS(ER)	AREA 86
KL=NX*NY	AREA 87
NHIT=0	AREA 88
C	AREA 89
C	AREA 90
PROCESS KL CELL'S IN GRID	AREA 91
C	AREA 92
DO 200 KK=NSTART,KL	

```

WB(1)=-CF*CA
WB(2)=-CF*SA
WB(3)=-SF
II=((KK-1)/NX)+1
J=KK-(II-1)*NX
C      COMPUTE COORDINATES OF GRID CELL IN GRID PLANE
CELLZ=.5*CELSIZ
V=FLOAT((NY/2)-(II)*CELSIZ)+CELLZ
VREF=V+CELLZ
H=FLOAT((NX/2)-(J)*CELSIZ)+CELLZ
HREF=H+CELLZ
IV=RAV*(-1)*10.
IH=RAV*(-1)*10.
IVIH=10*IH+IV
C      COMPUTE H,V AT RANDOM POINT IN GRID CELL
V=V+CELSIZ*FLOAT(IV)/10.+CELSIZ/20.
H=H+CELSIZ*FLOAT(IH)/10.+CELSIZ/20.
C      X,Y,Z IN COORDINATE SYSTEM OF VEHICLE
XBS(1)=XSHIFT-V*CA*SE-H*SA
XBS(2)=YSHIFT-V*SA*SE+H*CA
XBS(3)=ZSHIFT+V*CL
CALL IRPIC(WP)
XBS(1)=XBS(1)+WP(1)*1.0E-4
XBS(2)=XBS(2)+WP(2)*1.0E-4
XBS(3)=XBS(3)+WP(3)*1.0E-4
XB(1)=XBS(1)-LNGTH*WB(1)
XB(2)=XBS(2)-LNGTH*WB(2)
XB(3)=XBS(3)-LNGTH*WB(3)
IF(XB(3).LE.GROUND)GOTO 200
C
C      TRACE RAY TO FIRST COMPONENT HIT
C
IR=IRSTRT
NASC=-1
110 CALL G1(SI,IRPRIM,XP)
IF(ICERR.GE.NGIERR)RETURN
IF(IRPRIM.LI.0)GOTO 200
IF(NASC.LE.NRPP)IRPRIM=0
IF(IRPRIM.EQ.0)GOTO 200
LOC=LIREO+IRPRIM-1
CALL UN2(LOC,ICODE,IDENT)
IDENT=IDENT-1
IF(IDENT-(IDENT/10)*10.EQ.0)GOTO 120
IR=IRPRIM
GOTO 110
120 IF(ICODE.NE.0)GOTO 130
WRITE (6,909)
GOTO 200
130 LOC=AREA+ICODE-1
ASTER(LOC)=ASTER(LOC)+AREA
NHIT=NHIT+1
200 CONTINUE
C
C      PRINT RESULTS
C
WRITE (6,910)A,E
WRITE (6,911)CELSIZ, CELSIZ, CELLUN,AREAUN
WRITE (6,912)
SUMA=0.
DO 250 I=1,999

```

```

AREA 93
AREA 94
AREA 95
AREA 96
AREA 97
AREA 98
AREA 99
AREA 100
AREA 101
AREA 102
AREA 103
AREA 104
AREA 105
AREA 106
AREA 107
AREA 108
AREA 109
AREA 110
AREA 111
AREA 112
AREA 113
AREA 114
AREA 115
AREA 116
AREA 117
AREA 118
AREA 119
AREA 120
AREA 121
AREA 122
AREA 123
AREA 124
AREA 125
AREA 126
AREA 127
AREA 128
AREA 129
AREA 130
AREA 131
AREA 132
AREA 133
AREA 134
AREA 135
AREA 136
AREA 137
AREA 138
AREA 139
AREA 140
AREA 141
AREA 142
AREA 143
AREA 144
AREA 145
AREA 146
AREA 147
AREA 148
AREA 149
AREA 150
AREA 151
AREA 152

```

```

LOC=LAREA+1-1
IF(ASTER(LOC).EQ.0.)GOTO 250
WRITE (6,913)I,ASTER(LOC)
SUMA=SUMA+ASTER(LOC)
250 CONTINUE
WRITE (6,914)SUMA
WRITE (6,915)KL,NHIT
RETURN
END

SUBROUTINE G1(S1,IRPRIM,XP)
MAIN RAY TRACKING ROUTINE
GIVEN A RAY IN REGION IR AT POINT XB WITH DIRECTION
COSINES WB; FIND THE DISTANCE (S1) TO THE NEXT REGION
AND THE NUMBER OF THAT REGION (IRPRIM)

NASC=-2          $$ CALL FROM CALC TO FIND NORMAL DIST
NASC=-1          $$ START NEW RAY
IVOLUM=1         $$ CALL FROM VOLUM
ITESTG=1         $$ CALL FROM TESTG
DIST             $$ TOTAL DIST TRAVELED BY RAY SO FAR

DIMENSION MASTER(30000),XP(3),XBD(3),LSURT(50),NASC(50)
COMMON ASTER(30000)
COMMON/PAREM/XB(3),WB(3),IR
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
COMMON/UNCGM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
1 LDATA,LRLN,LROT,LLO,LOCDA,I15,I30,LBODY,NASC,KLOOP
COMMON/CAL/NIR,SIN,ANGLE,NTYPE,SSPACE,L,XS(3),WS(3),TRAVEL,
1 SN,V,H,IVIII
COMMON/WALT/LIRFO,NGIERR
COMMON/LSU/LSURF
COMMON/CONTRL/ITESTG,IRAYSK,IENLTV,IVOLUM,IWOT,ITAPE8,NO,IYES
COMMON/DAVIS/IGRID,LOOP,INORM
COMMON/CFL/CFLSIZ
EQUIVALENCE (ASTER,MASTER)

901 FORMAT(1H0,32HEXERRR IN G1 AT 140      BAD ITYPE,5X,4HITY=,15)
902 FORMAT(1H0,33HEXERRR IN G1 AT 510      SM 0= PINF,5X,3HIR=,15)
903 FORMAT(4H XB=,3E20.8/4H WB=,3E20.8/10X,5HKL00P,12X,3HNB0,
1 12X,3HLRI,12X,3HLRO,11X,4HNNHIT,11X,4HLOOP/6I15)
904 FORMAT(1H1,15(2H* ),3X, 9HEXERRR NO.,15,3X,15(2H* )//)
905 FORMAT(34X,4HCELL,2I4)
906 FORMAT(19H ERROR IN G1 AT 640//4H J1=,110,4H J2=,110,7H LSURF=,
1 110,6H NASC=,110,4H IR=,110/4H SM=,E21.10,4H S1=,E17.10/
2 4H WB=,3E21.10/4H XB=,3E21.10)
907 FORMAT(50H THE (SOLID POSITION/DEPTH/POINT NOW AT) IS ONE OF,
1 6H THESE/6H XBD =,3E21.10/6H DIST=,E21.10//)
908 FORMAT(4X,3HRIN,12X,4HROUT,7X,8HENTERING,2X,7HLEAVING,3X,
1 8HBODY NO.,5X,3HRAY,735X,8HSIDE NO.,2X,8HSIDE NO.//)
910 FORMAT(//16H TILT RIN=ROUT=,E20.10,30X,2HI=,15//)
911 FORMAT(2(2X,E15.8),4X,12,8X,12,6X,15,5X,7HSTARTED/)
912 FORMAT(2(2X,E15.8),4X,12,8X,12,6X,15,5X,7HHAS HIT/)
913 FORMAT(2(2X,E15.8),4X,12,8X,12,6X,15,5X,7HLEAVING/)
914 FORMAT(2(2X,E15.8),4X,12,8X,12,6X,15,5X,7H IN /)

```

```

AREA 153
AREA 154
AREA 155
AREA 156
AREA 157
AREA 158
AREA 159
AREA 160
AREA 161
AREA 162
AREA 163
AREA 164
AREA 165
**** 27
G1 2
G1 3
G1 4
G1 5
G1 6
G1 7
G1 8
G1 9
G1 10
G1 11
G1 12
G1 13
G1 14
G1 15
G1 16
G1 17
G1 18
G1 19
G1 20
G1 21
G1 22
G1 23
G1 24
G1 25
G1 26
G1 27
G1 28
G1 29
G1 30
G1 31
G1 32
G1 33
G1 34
G1 35
G1 36
G1 37
G1 38
G1 39
G1 40
G1 41
G1 42
G1 43
G1 44
G1 45
G1 46
G1 47

```



```

915 FORMAT(2(2X,E15.8),4X,12,8X,12,6X,15,5X,8HENTERING/)
916 FORMAT(2(2X,E15.8),4X,12,8X,12,6X,15,5X,8HWILL HIT/)
917 FORMAT(//4(14H END ERROR NO.,14,3X)/)
918 FORMAT(11H0,15,21H ERRORS IN G1, RETURN)

```

```

      INORM=0
      IF(NASC.EQ.-2) INORM=1
      SI=0.
      IF(NASC.GT.0) GOTO 20
      NEW RAY
      DIST=0.
      IF(KLOOP.LT.32000) GOTO 15
      KLOOP=0
      LION=LIO+NBODY+NRPP-1
      DO 10 I=LIO,LION
      MASTER(I)=0
10 CONTINUE
15 KLOOP=KLOOP+1

```

```

      BEGIN TRACING RAY

```

```

20 SM=PI*F
  NHIT=0
  LOC=LREGD+IR-1
  CALL UN2(LOC,LOC,NC)
  LOC=LIC-1

```

```

      NC=NUM OF BODIES IN REGION DESCRIPTION
      FIND RIN AND ROUT FOR EACH OF THESE BODIES
      RIN IS DIST FROM XB TO POINT WHERE RAY ENTERS THE BODY
      ROUT IS DIST FROM XB TO POINT WHERE RAY LEAVES THE BODY
      IF ROUT = -PI*F RAY DOES NOT HIT BODY
      G1 SELECTS SMALLEST OF RIN AND ROUT DISTANCES 0 DIST
      1) UNIQUE RIN VALUE - NEXT BODY IN PATH OF RAY
      2) 2 OR MORE RIN VALUES - 2 OR MORE BODIES HAVE
        A COMMON SURFACE
      3) ROUT FOR CURRENT BODY MEANS RAY WILL LEAVE
        THIS BODY BEFORE ENCOUNTERING ANOTHER

```

```

      DO 500 N=1,NC
      LOC=LIC+1
      CALL UN2(LOC,DUM,NBO)
      ITEMP=LIO+NBO-1
      CALL UN3(ITEMP,LRI,LRO,LOOP)
      ITEMP=LBOODY+3*(NBO-1)
      CALL UN2(ITEMP,ITYPF,LOCDA)
      IF(LOOP.NE.KLOOP) GOTO 130
      (CONTINUATION OF RAY
      IF(ITYPF.GT.11) GOTO 140
      IJK=LK17+NBO-1
      RIN=ASTER(IJK)
      IJK=LROT+NBO-1
      ROUT=ASTER(IJK)
      IF(ITYPF.LT.10) GOTO 320
      FOR AND ARS
      IF 91ST .GE. ROUT COMPUTE RIN / ROUT SET
      IF(ROUT.LT.0.) GOTO 320
      IF(DIST.LT.ROUT) GOTO 320
      IF(NASC.EQ.NBO) NASC=0

```

G1	48
G1	49
G1	50
G1	51
G1	52
G1	53
G1	54
G1	55
G1	56
G1	57
G1	58
G1	59
G1	60
G1	61
G1	62
G1	63
G1	64
G1	65
G1	66
G1	67
G1	68
G1	69
G1	70
G1	71
G1	72
G1	73
G1	74
G1	75
G1	76
G1	77
G1	78
G1	79
G1	80
G1	81
G1	82
G1	83
G1	84
G1	85
G1	86
G1	87
G1	88
G1	89
G1	90
G1	91
G1	92
G1	93
G1	94
G1	95
G1	96
G1	97
G1	98
G1	99
G1	100
G1	101
G1	102
G1	103
G1	104
G1	105
G1	106
G1	107

130	LRI=1	G1	108
	LRO=1	G1	109
	ITY=ITYE+1	G1	110
	IF(ITY.GE.1.AND.ITY.LE.12)GOTO 200	G1	111
140	IERR=IERR+1	G1	112
	WRITE (6,901)ITYE	G1	113
	GOTO 800	G1	114
C		G1	115
C	RPP BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	G1	116
200	GOTO(205,210,215,220,225,230,235,240,245,250,255,260),ITY	G1	117
205	CALL RPP(NB0)	G1	118
	GOTO 300	G1	119
210	CALL ROX	G1	120
	GOTO 300	G1	121
215	CALL SPH	G1	122
	GOTO 300	G1	123
220	CALL RCC	G1	124
	GOTO 300	G1	125
225	CALL REC	G1	126
	GOTO 300	G1	127
230	CALL TRC	G1	128
	GOTO 300	G1	129
235	CALL ELL	G1	130
	GOTO 300	G1	131
240	CALL RAW	G1	132
	GOTO 300	G1	133
245	CALL ARB	G1	134
	GOTO 300	G1	135
250	CALL TEC	G1	136
	GOTO 300	G1	137
255	CALL TOR	G1	138
	GOTO 300	G1	139
260	CALL ARS	G1	140
C		G1	141
300	IJK=LRI+NBO-1	G1	142
	ASTER(IJK)=RIN	G1	143
	IJK=LROT+NBO-1	G1	144
	ASTER(IJK)=ROUT	G1	145
	IJK=LIO+NBO-1	G1	146
	MASTER(IJK)=KLOOP+115*(LRO+64*LRI)	G1	147
C		G1	148
320	IF(NASC.NE.NBO)GOTO 330	G1	149
	IF(LSURF)500,500,340	G1	150
C		G1	151
330	IF(ROUT.LE.0.)GOTO 500	G1	152
	IF(RIN.GT.0.)GOTO 350	G1	153
C		G1	154
340	IF(ABS(ROUT-SM).GT.SM*1.0E-6)GOTO 341	G1	155
	ROUT=SM	G1	156
	IJK=LROT+NBO-1	G1	157
	ASTER(IJK)=ROUT	G1	158
	GOTO 345	G1	159
341	IF(ROUT-SM)342,345,500	G1	160
342	IF(DIST.GE.ROUT)GOTO 500	G1	161
	NHIT=0	G1	162
345	NHIT=NHIT+1	G1	163
	SM=ROUT	G1	164
	LSURT(NHIT)=-LRO	G1	165
	NASC1(NHIT)=NBO	G1	166
	GOTO 500	G1	167

C		G1	168
	350 IF (ABS(RIN-SM).GT.SM*1.0E-6)GOTO 351	G1	169
	RIN=SM	G1	170
	IJK=LRIN+NBO-1	G1	171
	ASTER(IJK)=RIN	G1	172
	GOTO 355	G1	173
	351 IF (RIN-SM) 352,355,500	G1	174
	352 IF (DIST.GE.RIN)GOTO 340	G1	175
	NHIT=0	G1	176
	355 NHIT=NHIT+1	G1	177
	SM=RIN	G1	178
	LSURT(NHIT)=LRI	G1	179
	NASCT(NHIT)=NBO	G1	180
C		G1	181
	500 CONTINUE	G1	182
C		G1	183
C	SM.GE.PINF ERROR AT 510 IN G1	G1	184
C		G1	185
	IF (SM.LT.PINF)GOTO 530	G1	186
	WRITE (6,902)IR	G1	187
	WRITE (6,903)XB,WB,KLOOP,NBO,LRI,LRO,NHIT,LOOP	G1	188
	GOTO 700	G1	189
C		G1	190
	530 S1=S1+SM-DIST	G1	191
	DIST=SM	G1	192
	XP(1)=XB(1)+SM*WB(1)	G1	193
	XP(2)=XB(2)+SM*WB(2)	G1	194
	XP(3)=XB(3)+SM*WB(3)	G1	195
C		G1	196
	IF (NASC.EQ.-2)RETURN	G1	197
C		G1	198
C	FIND NEXT REGION (IRPRIM)	G1	199
C		G1	200
	DO 640 NN=1,NHIT	G1	201
	NASC =NASCT(NN)	G1	202
	LSURF=LSURT(NN)	G1	203
	LTRUE=0	G1	204
	LOC=LBODY+3*(NASC-1)	G1	205
	LOC=LOC+1	G1	206
	CALL UN2(LOC,LENT,LEAV)	G1	207
	LOC=LOC+1	G1	208
	CALL UN2(LOC,NENT,NEAV)	G1	209
	IF (LSURF.LE.0)GOTO 600	G1	210
	J1=LENT	G1	211
	J2=LENT+NENT-1	G1	212
	GOTO 610	G1	213
	600 J1=LEAV	G1	214
	J2=LEAV+NEAV-1	G1	215
C		G1	216
	610 IRPRIM=MASTER(J2)	G1	217
	IF (J1.LE.J2)GOTO 620	G1	218
	IF (NASC.GT.NRPP)GOTO 700	G1	219
	IF (LSURF)630,700,700	G1	220
C		G1	221
	620 DO 625 J=J1,J2	G1	222
	IRPRIM=MASTER(J)	G1	223
	CALL WOWI(IRPRIM,LSURF,NASC,LTRUE)	G1	224
	IF (LTRUE.GT.0)GOTO 650	G1	225
	625 CONTINUE	G1	226
C		G1	227

C	RPP CHECK	G1	228
C	IF(NASC.GT.NRPP)GOTO 640	G1	229
	IF(LSURF)630,700,640	G1	230
630	CALL RPP2(LSURF,XP,IRP)	G1	231
	IF(IRP.GT.0)GOTO 631	G1	232
	IRPRIM=0	G1	233
	RETURN	G1	234
C		G1	235
631	LTRUE=0	G1	236
	LOC=LBODY+3*(IRP-1)	G1	237
	LOC=LOC+1	G1	238
	CALL UN2(LOC,LENT,LEAV)	G1	239
	LOC=LOC+1	G1	240
	CALL UN2(LOC,NENT,NEAV)	G1	241
	J1=LENT	G1	242
	J2=LENT+NENT-1	G1	243
	IF(J1.GT.J2)GOTO 700	G1	244
C		G1	245
	DO 632 J=J1,J2	G1	246
	IRPRIM=MASTER(J)	G1	247
	CALL WOWI(IRPRIM,LSURF,IRP,LTRUE)	G1	248
	IF(LTRUE.GT.0)GOTO 650	G1	249
632	CONTINUE	G1	250
C		G1	251
640	CONTINUE	G1	252
	GOTO 700	G1	253
C		G1	254
C	NEXT REGION (IRPRIM) HAS BEEN DETERMINED	G1	255
C		G1	256
650	IF(IR.EQ.IRPRIM)GOTO 660	G1	257
	IF(S1.EQ.0.)GOTO 660	G1	258
	IF(S1.LT.0.)GOTO 700	G1	259
	IF(ABS(S1).LE.1.0E-6)GOTO 660	G1	260
	IF(IVOLUM.EQ.IYES)RETURN	G1	261
	IF(ITESTG.EQ.IYES)RETURN	G1	262
	LOC=LIRFO+IR-1	G1	263
	CALL UN2(LOC,ICODE,IDENT)	G1	264
	LOC=LIRFO+IRPRIM-1	G1	265
	CALL UN2(LOC,ICODE1,IDENT1)	G1	266
	IF(IDENT.EQ.1)GOTO 655	G1	267
	IF(IDENT.EQ.IDENT1)GOTO 660	G1	268
	RETURN	G1	269
655	IF(ICODE.NE.ICODE1)RETURN	G1	270
660	IR=IRPRIM	G1	271
	GOTO 20	G1	272
C		G1	273
C	DIAGNOSTIC ERROR PRINT	G1	274
C		G1	275
700	IERR=IERR+1	G1	276
	WRITE (6,904)IERR	G1	277
	IF(IVOLUM.EQ.IYES.OR.ITESTG.EQ.IYES)GOTO 705	G1	278
	IH=ABS(H/CELSIZ)+.5	G1	279
	IF(H.LT.0.)IH=-IH	G1	280
	IV=ABS(V/CELSIZ)+.5	G1	281
	IF(V.LT.0.)IV=-IV	G1	282
	WRITE (6,905)IH,IV	G1	283
705	WRITE (6,906)J1,J2,LSURF,NASC,IR,SM,S1,WB,XB	G1	284
	XBD(1)=XB(1)-DIST	G1	285
	XBD(2)=XB(2)-DIST	G1	286
		G1	287

XBD(3)=XB(3)-DIST	G1	288
WRITE (6,907)XBD,DIST	G1	289
WRITE (6,908)	G1	290
NN=NBODY+NRPP	G1	291
C	G1	292
DO 750 I=1,NN	G1	293
LOC=LIO+I-1	G1	294
CALL UN3(LOC,I1,I2,I3)	G1	295
IF(KLOOP.NE.I3)GOTO 750	G1	296
IJK=LRIN+I-1	G1	297
RIN=ASTER(IJK)	G1	298
IJK=LROUT+I-1	G1	299
ROUT=ASTER(IJK)	G1	300
IF(RIN.NE.ROUT)GOTO 710	G1	301
WRITE (6,910)RIN,I	G1	302
GOTO 750	G1	303
C	G1	304
710 IF(ABS(RIN).NE.PINF)GOTO 720	G1	305
IF(ABS(ROUT)-PINF)740,750,740	G1	306
720 IF(RIN-DIST)730,744,745	G1	307
730 IF(ROUT-DIST)741,742,743	G1	308
C	G1	309
740 WRITE (6,911)RIN,ROUT,I1,I2,I	G1	310
GOTO 750	G1	311
741 WRITE (6,912)RIN,ROUT,I1,I2,I	G1	312
GOTO 750	G1	313
742 WRITE (6,913)RIN,ROUT,I1,I2,I	G1	314
GOTO 750	G1	315
743 WRITE (6,914)RIN,ROUT,I1,I2,I	G1	316
GOTO 750	G1	317
744 WRITE (6,915)RIN,ROUT,I1,I2,I	G1	318
GOTO 750	G1	319
745 WRITE (6,916)RIN,ROUT,I1,I2,I	G1	320
C	G1	321
750 CONTINUE	G1	322
WRITE (6,917)IERR,IERR,IERR,IERR	G1	323
IRPRIM=-1	G1	324
C	G1	325
800 IF(IERR.GE.NGIERR)WRITE (6,918)NGIERR	G1	326
RETURN	G1	327
END	G1	328
C	G1	329
C	G1	330
SUBROUTINE WOWI(JREG,LSURF,NEX,LTRUE)	****	28
C	WOWI	2
C GIVEN A POINT (XB) AND A REGION (JREG), DOES XB	WOWI	3
C LIE WITHIN JREG	WOWI	4
C	WOWI	5
C SUFFICIENT CONDITION FOR POINT XB TO BE IN REGION	WOWI	6
C JREG, IS THAT REGION DESCRIPTION OF JREG BE	WOWI	7
C SATISFIED. TWO REGIONS CANNOT BE SATISFIED FOR	WOWI	8
C THE SAME POINT	WOWI	9
C	WOWI	10
C + OPERATOR VALID IF ROUT.GT.0 AND RIN.LE.DIST.LT.ROUT	WOWI	11
C - OPERATOR VALID IF ROUT.LE.0 OR DIST.LT.RIN OR DIST.GE.ROUT	WOWI	12
C OR OPERATOR VALID IF ALL (+) AND (-) IN (OR) STATEMENT VALID	WOWI	13
C	WOWI	14
C REGION DESCRIPTION WITH 1 OR MORE (OR) STATEMENTS VALID	WOWI	15
C IF ANY ONE OF (OR) STATEMENTS IS VALID	WOWI	16
C REGION DESCRIPTION WITH NO (OR) STATEMENTS IS VALID ONLY	WOWI	17

C	IF EVERY (+) AND (-) OPERATOR IS VALID	WOWI	18
C		WOWI	19
	DIMENSION MASTER(30000)	WOWI	20
	COMMON ASTER(30000)	WOWI	21
	COMMON/PAREM/XB(3),WB(3),IR	WOWI	22
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	WOWI	23
	COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	WOWI	24
	1 LDATA,LRIN,LROT,LIO,LCDA,I15,I30,LBODY,NASC,KLOOP	WOWI	25
	EQUIVALENCE(ASTER,MASTER)	WOWI	26
C		WOWI	27
	901 FORMAT(1H0,32HERROR IN G1 AT 140 BAD ITYPE,5X,4HITY=,15)	WOWI	28
C		WOWI	29
	LOC=LREGD+JREG-1	WOWI	30
	CALL UN2(LOC,LOC,NC)	WOWI	31
	CALL UN2(LOC,IOP,NBO)	WOWI	32
	N=1	WOWI	33
	IOPER=IOP	WOWI	34
C		WOWI	35
C	EXAMINE NC CHOICES N=1,NC	WOWI	36
C		WOWI	37
	10 ITEMP=LIO+NBO-1	WOWI	38
	CALL UN3(ITEMP,LRI,LRO,LOOP)	WOWI	39
	ITEMP=LBODY+3*(NBO-1)	WOWI	40
	CALL UN2(ITEMP,ITYPE,LOCDA)	WOWI	41
	IF(LOOP.NE.KLOOP)GOTO 30	WOWI	42
C	CONTINUATION OF RAY	WOWI	43
	IF(ITYPE.GT.11)GOTO 40	WOWI	44
	IJK=LRIN+NBO-1	WOWI	45
	RIN=ASTER(IJK)	WOWI	46
	IJK=LROI+NBO-1	WOWI	47
	ROUT=ASTER(IJK)	WOWI	48
	IF(ITYPE.LT.10)GOTO 310	WOWI	49
C	TOR AND ARS	WOWI	50
C	IF DIST 0 ROUT COMPUT RIN/ROUT SET	WOWI	51
	IF(ROUT.LT.0.)GOTO 400	WOWI	52
	IF(DIST.LE.ROUT)GOTO 310	WOWI	53
C		WOWI	54
	30 LRI=1	WOWI	55
	LRO=1	WOWI	56
	ITY=ITYPE+1	WOWI	57
	IF(ITY.GE.1.AND.ITY.LE.12)GOTO 100	WOWI	58
	40 IERR=IERR+1	WOWI	59
	WRITE (6,901)ITYPE	WOWI	60
	RETURN	WOWI	61
C	RPP BOX SPH RCC REC TRC ELL RAW ARB TEC TOR ARS	WOWI	62
	100 GOTO(110,120,130,140,150,160,170,180,190,200,210,220),ITY	WOWI	63
	110 CALL RPP(NBO)	WOWI	64
	GOTO 300	WOWI	65
	120 CALL BOX	WOWI	66
	GOTO 300	WOWI	67
	130 CALL SPH	WOWI	68
	GOTO 300	WOWI	69
	140 CALL RCC	WOWI	70
	GOTO 300	WOWI	71
	150 CALL REC	WOWI	72
	GOTO 300	WOWI	73
	160 CALL TRC	WOWI	74
	GOTO 300	WOWI	75
	170 CALL ELL	WOWI	76
	GOTO 300	WOWI	77

180 CALL RAW	WOWI 78
GOTO 300	WOWI 79
190 CALL ARR	WOWI 80
GOTO 300	WOWI 81
200 CALL TEC	WOWI 82
GOTO 300	WOWI 83
210 CALL TOR	WOWI 84
GOTO 300	WOWI 85
220 CALL ARS	WOWI 86
C	WOWI 87
300 IJK=LIO+NBO-1	WOWI 88
MASTER(IJK)=KLOOP+I15*(LRO+64*LR1)	WOWI 89
C	WOWI 90
310 IF(ROUT.LL.0.)GOTO 330	WOWI 91
IF(ABS(RIN-DIST).GT.DIST*1.0E-6)GOTO 320	WOWI 92
RIN=DIST	WOWI 93
GOTO 330	WOWI 94
C	WOWI 95
320 IF(ABS(ROUT-DIST).LE.DIST*1.0E-6)ROUT=DIST	WOWI 96
C	WOWI 97
330 IJK=LIN+NBO-1	WOWI 98
ASTER(IJK)=RIN	WOWI 99
IJK=LROT+NBO-1	WOWI 100
ASTER(IJK)=ROUT	WOWI 101
C	WOWI 102
C TEST CONDITIONS FOR XB IN JREG (LTRUE SET=1)	WOWI 103
C	WOWI 104
400 IF(IOPER.GT.4)GOTO 500	WOWI 105
C (+) OPERATOR TEST FOR INSIDE RIN.LE.DIST.LT.ROUT	WOWI 106
IF(RIN.GT.DIST)GOTO 700	WOWI 107
IF(DIST-ROUT)600,700,700	WOWI 108
C (-) OPERATOR TEST FOR OUTSIDE DIST.LT.RIN DIST.GE.ROUT	WOWI 109
500 IF(ROUT.LE.0.)GOTO 600	WOWI 110
IF(DIST.LT.RIN)GOTO 600	WOWI 111
IF(DIST.EQ.RIN)GOTO 700	WOWI 112
IF(DIST.LT.ROUT)GOTO 700	WOWI 113
C CHECK NEXT BODY IN DESCRIPTION	WOWI 114
600 IF(N.GE.NC)GOTO 800	WOWI 115
N=N+1	WOWI 116
LOCD=LOCD+1	WOWI 117
CALL UN2(LOCD,IOPER,NBO)	WOWI 118
IF(IOPER.EQ.1.OR.IOPER.EQ.5)GOTO 800	WOWI 119
GOTO 10	WOWI 120
C OR OPERATOR	WOWI 121
700 IF(1OP.NE.1.AND.1OP.NE.5) RETURN	WOWI 122
IF(N.GE.NC)RETURN	WOWI 123
N=N+1	WOWI 124
DO 710 NN=N,NC	WOWI 125
LOCD=LOCD+1	WOWI 126
CALL UN2(LOCD,IOPER,NBO)	WOWI 127
IF(IOPER.NE.1.AND.IOPER.NE.5)GOTO 710	WOWI 128
N=NN	WOWI 129
GOTO 10	WOWI 130
710 CONTINUE	WOWI 131
RETURN	WOWI 132
C	WOWI 133
800 LTRUE=LTRUE+1	WOWI 134
RETURN	WOWI 135
END	WOWI 136
C	WOWI 137

C	SUBROUTINE ARB	WOWI 138
	DIMENSION AA(6,4),XP(3)	**** 29
	COMMON ASTER(30000)	ARB 2
	COMMON/PAREM/XB(3),WB(3),IR	ARB 3
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	ARB 4
	COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	ARB 5
	1 LDATA,LRI,N,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	ARB 6
C	LOC=LOCDA-1	ARB 7
	DO 10 I=1,6	ARB 8
	LOC=LOC+1	ARB 9
	CALL UN2(LOC,LD,LC)	ARB 10
	AA(I,1)=ASTER(LC)	ARB 11
	AA(I,2)=ASTER(LC+1)	ARB 12
	AA(I,3)=ASTER(LC+2)	ARB 13
	AA(I,4)=ASTER(LD)	ARB 14
10	CONTINUE	ARB 15
	RIN=-PINF	ARB 16
	ROUT=PINF	ARB 17
	LRO=0	ARB 18
	LRI=0	ARB 19
	S1=0.	ARB 20
	S2=0.	ARB 21
	L1=0	ARB 22
	L2=0	ARB 23
	DO 70 I=1,6	ARB 24
	D=AA(I,4)	ARB 25
	SNUM=-D-AA(I,1)*XB(1)-AA(I,2)*XB(2)-AA(I,3)*XB(3)	ARB 26
	SDEN=AA(I,1)*WB(1)+AA(I,2)*WB(2)+AA(I,3)*WB(3)	ARB 27
	IF(SDEN)20,70,30	ARB 28
20	IF(SNUM)40,70,70	ARB 29
30	IF(SNUM)70,70,40	ARB 30
40	S=SNUM/SDEN	ARB 31
	DO 50 K=1,3	ARB 32
	XP(K)=XB(K)+S*WB(K)	ARB 33
50	CONTINUE	ARB 34
	DO 60 J=1,6	ARB 35
	IF(I.EQ.J)GOTO 60	ARB 36
	T=AA(J,1)*XP(1)+AA(J,2)*XP(2)+AA(J,3)*XP(3)+AA(J,4)	ARB 37
	IF(ABS(T).LE.1.0E-6)T=0.	ARB 38
	IF(T.LT.0.)GOTO 70	ARB 39
60	CONTINUE	ARB 40
	IF(L1.GT.0)GOTO 65	ARB 41
	L1=1	ARB 42
	S1=S	ARB 43
	GOTO 70	ARB 44
65	IF(ABS(S1-S).GT.1.0E-6)GOTO 100	ARB 45
70	CONTINUE	ARB 46
C	IF(L1)200,200,150	ARB 47
100	S2=S	ARB 48
	L2=1	ARB 49
	IF(ABS(S1-S2).LE.S1*1.0E-5)GOTO 200	ARB 50
	IF(S1-S2)110,200,120	ARB 51
110	RIN=S1	ARB 52
	ROUT=S2	ARB 53
	LRI=L1	ARB 54
	LRO=L2	ARB 55
	RETURN	ARB 56
		ARB 57
		ARB 58
		ARB 59

120	RIN=S?		ARB	60
	LRI=L2		ARB	61
130	ROUT=SI		ARB	62
	LRO=L1		ARB	63
	RETURN		ARB	64
150	DO 160 J=1,6		ARB	65
	IF(L1.EQ.J)GOTO 160		ARB	66
	T1=AA(J,1)*XB(1)+AA(J,2)*XB(2)+AA(J,3)*XB(3)+AA(J,4)		ARK	67
	IF(ABS(T1).LE.1.OE-6)T1=0.		ARB	68
	IF(T1.L1.0.)GOTO 200		ARB	69
160	CONTINUE		ARB	70
	GOTO 130		ARB	71
C			ARB	72
200	RIN=PINF		ARB	73
	ROUT=-PINF		ARB	74
	LRI=0		ARB	75
	LRO=0		ARB	76
	RETURN		ARB	77
	END		ARB	78
C			ARB	79
C			ARB	80
	SUBROUTINE ARS		****	30
	DIMENSION MASTER(30000),COL1(3),COL2(3),COL3(3),COL4(3),		ARS	2
1	U(3),V(3),W(3),SAVE(84)		ARS	3
	COMMON ASTER(30000)		ARS	4
	COMMON/PAREM/XB(3),WB(3),IR		ARS	5
	COMMON/GFCM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST		ARS	6
	COMMON/UNCDEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,		ARS	7
1	LDATA,LKIN,LROT,LIO,LOCDA,I15,I30,LBOUY,NASC,KLOOP		ARS	8
	COMMON/DAVIS/IGRID,LOOP,INORM		ARS	9
	EQUIVALENCE(COL11,COL1(1)),(COL12,COL1(2)),(COL13,COL1(3))		ARS	10
	EQUIVALENCE(COL21,COL2(1)),(COL22,COL2(2)),(COL23,COL2(3))		ARS	11
	EQUIVALENCE(COL31,COL3(1)),(COL32,COL3(2)),(COL33,COL3(3))		ARS	12
	EQUIVALENCE(COL41,COL4(1)),(COL42,COL4(2)),(COL43,COL4(3))		ARS	13
	EQUIVALENCE(ASTER,MASTER)		ARS	14
C			ARS	15
901	FORMAT(IHO,2IHTRUGBLE IN ARS AT 150)		ARS	16
902	FORMAT(IHO,48HPOSSIBLE ERROR IN ARBITRARY SURFACE, CHECK INPUT,I5)		ARS	17
C			ARS	18
C			ARS	19
C	LOCDA+0	ASTER/MASTER	TEMPORARY STORAGE	ARS
C	+1	M	NO. CURVES	ARS
C	+2	N	NO. POINTS/CURVE	ARS
C	+3	IGDTL	GRID TOLERANCE	ARS
C	+4	BIAS	NO. OF NEGATIVE OR ZERO HITS	ARS
C	+5	XB(X)		ARS
C	+6	XB(Y)		ARS
C	+7	XB(Z)		ARS
C	+8	(84 WORDS)	RESERVED FOR HITS	ARS
C	.	.		ARS
C	.	.		ARS
C	.	.		ARS
C	+91	.		ARS
C	+92	X)		ARS
C	.	Y)N=1)	M SETS OF N POINTS	ARS
C	.	Z))M=1		ARS
C	.	K)		ARS
C	.	.)		ARS
C	.	.)N=2)		ARS
C	.	.)		ARS

C	.	.))	ARS	40
C					ARS	41
C	M	=	THE	NUMBER	OF	CURVES
C	N	=	THE	NUMBER	POINTS/CURVE	
C	IGDTL	=	THE	NUMBER	OF	GRID
C	IBIAS	=	THE	INDEX	INTO	MASTER
C					TO	THE
C					NUMBER	OF
C					DISCARDED	HITS
C						
C	LOCPTS	IS	THE	LOCATION	OF	THE
C					AREA	IN
C					MASTER-ASTER	
C					RESERVED	FOR
C					STORING	HITS
C						
C	LOCARY	IS	THE	LOCATION	IN	MASTER-ASTER
C					OF	THE
C					DATA	POINTS
C					THEMSELVES	IN
C					FORMAT(X , Y , Z ,
C					GRID	SQUARE)
C						
	NE	=	4			
	IWH	=	MASTER(LOCDA)			
	IBIAS	=	IWH+4			
	LOCPTS	=	IBIAS+4			
	IF(INORM.EQ.0)	GOTO	20			
	DO	10	I=1,84			
	IJK	=	LOCPTS+I-1			
	SAVE(I)	=	ASTER(IJK)			
10	CONTINUE					
	GOTO	30				
20	ASTER(IWH+5)	=	XB(1)			
	ASTER(IWH+6)	=	XB(2)			
	ASTER(IWH+7)	=	XB(3)			
30	IF(KLOOP.EQ.LOOP)	GOTO	400			
C						
	LRI	=	1			
	LRO	=	1			
	M	=	MASTER(IWH+1)			
	N	=	MASTER(IWH+2)			
	IGDTL	=	MASTER(IWH+3)			
	LOCARY	=	LOCPTS+21*NE			
	NHITS	=	0			
	ASTER(LOCPTS)	=	PINF			
	MASTER(IBIAS)	=	0			
	KAPPA	=	M-1			
C						
C						
C	IN	THE	EVEN	CASE,	THE	TRIANGLES
C					ARE	-
C						
C	(1)	(I,J)	(I+1,J)	(I,J+1)		
C	(2)			(I,J-1)		
C	(3)		(I-1,J)	(I,J+1)		
C	(4)			(I,J-1)		
C						
C						
C	IN	THE	ODD	CASE,	THE	TRIANGLES
C					ARE	-
C						
C	(1)	(I,J)	(I+1,J)	(I+1,J+1)		
C	(2)			(I+1,J-1)		
C	(3)		(I-1,J)	(I+1,J+1)		
C	(4)			(I+1,J-1)		
C						
C	NOTE	THAT	THE	ONLY	DIFFERENCE	IS
C					THE	ROW
C					DESIGNATION	OF
C					W	
C	BECAUSE	OF	THE	INCREMENTATION	OF	I
C					AND	J,
C					WE	NEED
C					CONSIDER	ONLY
C					CASES	(1)
					AND	(2)

C	DO 200 I=1,KAPPA	ARS 100
	DO 200 J=1,N	ARS 101
	ITRY=0	ARS 102
	K=(I+J)/2	ARS 103
	IODD=I+J-2*K	ARS 104
C	IW1=4*((I-1+IODD)*N+J)+LOCARY	ARS 105
	IF(N.LE.J)GOTO 190	ARS 106
100	IV1=4*(I*N+(J-1))+LOCARY	ARS 107
	IU1=4*((I-1)*N+(J-1))+LOCARY	ARS 108
	IF(INORM.EQ.0)GOTO 110	ARS 109
	IF(IABS(IGRID-MASTER(IU1+3)).GT.IGDTL)GOTO 200	ARS 110
	IF(IABS(IGRID-MASTER(IV1+3)).GT.IGDTL)GOTO 200	ARS 111
	IF(IABS(IGRID-MASTER(IW1+3)).GT.IGDTL)GOTO 180	ARS 112
110	DO 115 K=1,3	ARS 113
	IJK=IU1+K-1	ARS 114
	U(K)=ASTER(IJK)	ARS 115
	IJK=IV1+K-1	ARS 116
	V(K)=ASTER(IJK)	ARS 117
	IJK=IW1+K-1	ARS 118
	W(K)=ASTER(IJK)	ARS 119
115	CONTINUE	ARS 120
C		ARS 121
C	AT THIS TIME WE HAVE U,V,W SIDES OF TRIANGLE	ARS 122
C		ARS 123
	DO 120 K=1,3	ARS 124
	COL1(K)=U(K)-W(K)	ARS 125
	COL2(K)=V(K)-W(K)	ARS 126
	COL3(K)=-W(K)	ARS 127
	COL4(K)=XR(K)-W(K)	ARS 128
120	CONTINUE	ARS 129
	0 = COL11*(COL22*COL33-COL23*COL32)	ARS 130
	1 -COL12*(COL21*COL33-COL23*COL31)	ARS 131
	2 +COL13*(COL21*COL32-COL22*COL31)	ARS 132
	IF(ABS(U).LE.1.0E-6)GOTO 180	ARS 133
C		ARS 134
	DALPHA= COL41*(COL22*COL33-COL23*COL32)	ARS 135
	1 -COL42*(COL21*COL33-COL23*COL31)	ARS 136
	2 +COL43*(COL21*COL32-COL22*COL31)	ARS 137
	ALPHA=DALPHA/D	ARS 138
	IF(ABS(ALPHA*(1.-ALPHA)).LT.0.)GOTO 180	ARS 139
C		ARS 140
	DBETA = COL11*(COL42*COL33-COL43*COL32)	ARS 141
	1 -COL12*(COL41*COL33-COL43*COL31)	ARS 142
	2 +COL13*(COL41*COL32-COL42*COL31)	ARS 143
	BETA=DBETA/D	ARS 144
	IF(ABS(BETA*(1.-BETA)).LT.0.)GOTO 180	ARS 145
	TP=ALPHA+BETA	ARS 146
	IF(ABS(TP*(1.-TP)).LT.0.)GOTO 180	ARS 147
C		ARS 148
	DS = COL11*(COL22*COL43-COL23*COL42)	ARS 149
	1 -COL12*(COL21*COL43-COL23*COL41)	ARS 150
	2 +COL13*(COL21*COL42-COL22*COL41)	ARS 151
	S=DS/D	ARS 152
C		ARS 153
	IF(NHITS.GT.20)GOTO 400	ARS 154
	LIMIT=NHITS+1	ARS 155
	LIMIT1=LOCHTS+20*NE-1	ARS 156
	TRY=1-ITRY-ITRY	ARS 157
		ARS 158
		ARS 159

CALL CROSS(COL3,COL1,COL2)	ARS 160
DO 140 L=1,3	ARS 161
COL3(L)=TRY*COL3(L)	ARS 162
140 CONTINUE	ARS 163
DO 150 L=1,LIMIT	ARS 164
INDEX=LOCHTS+(L-1)*NE	ARS 165
IF(S.LE.ASTER(INDEX))GOTO 160	ARS 166
150 CONTINUE	ARS 167
WRITE (6,901)	ARS 168
GOTO 180	ARS 169
C	ARS 170
C	ARS 171
160 DO 165 L=INDEX,LIMIT1	ARS 172
IJK=LIMIT1+INDEX-L	ARS 173
IJK1=IJK+NE	ARS 174
ASTER(IJK1)=ASTER(IJK)	ARS 175
165 CONTINUE	ARS 176
ASTER(INDEX)=S	ARS 177
DO 170 L=1,3	ARS 178
IJK=INDEX+L	ARS 179
ASTER(IJK)=COL3(L)	ARS 180
170 CONTINUE	ARS 181
NHITS=NHITS+1	ARS 182
180 IF(I TRY.GT.0)GOTO 200	ARS 183
190 IW1=IW1-8	ARS 184
I TRY=1	ARS 185
IF(J.GT.1)GOTO 100	ARS 186
200 CONTINUE	ARS 187
C	ARS 188
C THIS SECTION CHECKS FOR PROPER ENTER-LEAVE SEQUENCE IN HITS TABLE	ARS 189
C	ARS 190
IF(NHITS-1)800,210,220	ARS 191
210 ASTER(LOCHTS)=PINF	ARS 192
IJK=LOCHTS+NE	ARS 193
ASTER(IJK)=PINF	ARS 194
GOTO 800	ARS 195
220 ILEAVE=1	ARS 196
SLAST=-PINF	ARS 197
C	ARS 198
C ILEAVE = -1 IMPLIES AN ENTRY	ARS 199
C ILEAVE = +1 IMPLIES AN EXIT	ARS 200
C ENTRIES AND EXITS SHOULD ALTERNATE IN TABLE	ARS 201
C	ARS 202
DO 300 L=1,NHITS	ARS 203
INDEX=LOCHTS+(L-1)*NE	ARS 204
DO 230 L1=1,3	ARS 205
IJK=INDEX+L1	ARS 206
COL4(L1)=ASTER(IJK)	ARS 207
230 CONTINUE	ARS 208
TEMP=DOT(WR,COL4)	ARS 209
INEXT=SIGN(1.0,TEMP)	ARS 210
IF(ABS(SLAST-ASTER(INDEX)).GT.1.0E-7)GOTO 235	ARS 211
IF(ILEAVE*INEXT.GE.0)GOTO 260	ARS 212
L TRY=L	ARS 213
INDEX=INDEX-NE	ARS 214
GOTO 270	ARS 215
235 IJK=INDEX+NE	ARS 216
IF(ABS(ASTER(INDEX)-ASTER(IJK)).GT.1.0E-7)GOTO 240	ARS 217
IF(ILEAVE*INEXT)290,250,250	ARS 218
240 IF(ILEAVE*INEXT)290,280,280	ARS 219

C		ARS	220
C	BAD START OF A NEW S SET - TRY TO FIND AN ALTERNATING MEMBER	ARS	221
C		ARS	222
	250 LTRY=L	ARS	223
	251 LTRY=LTRY+1	ARS	224
	IF(LTRY.GT.NHITS)GOTO 280	ARS	225
	INDEX1=LOCHTS+(LTRY-1)*NE	ARS	226
	IF(ABS(ASTER(INDEX)-ASTER(INDEX1)).GT.1.0E-7)GOTO 280	ARS	227
	DO 252 L1=1,3	ARS	228
	IJK=INDEX1+L1	ARS	229
	COL4(L1)=ASTER(IJK)	ARS	230
	252 CONTINUE	ARS	231
	TEMP=DOT(WB, COL4)	ARS	232
	INEXT=SIGN(1.0, TEMP)	ARS	233
	IF(ILEAVE*INEXT.GE.0)GOTO 251	ARS	234
	LTRY=L+1	ARS	235
	GOTO 270	ARS	236
C		ARS	237
C	AT THIS POINT WE HAVE DETECTED TWO CONSECUTIVE ENTRIES OR EXITS	ARS	238
C	TRY TO RESOLVE BY DELETING ITEMS WITH EQUAL S ENTRIES	ARS	239
C		ARS	240
	260 LTRY=L	ARS	241
	261 LTRY=LTRY+1	ARS	242
	IF(LTRY.LE.NHITS)GOTO 262	ARS	243
	LTRY=L+1	ARS	244
	GOTO 270	ARS	245
	262 INDEX1=LOCHTS+(LTRY-1)*NE	ARS	246
	IF(ABS(ASTER(INDEX)-ASTER(INDEX1)).LE.1.0E-7)GOTO 263	ARS	247
	LTRY=L+1	ARS	248
	GOTO 270	ARS	249
	263 DO 264 L1=1,3	ARS	250
	IJK=INDEX1+L1	ARS	251
	COL4(L1)=ASTER(IJK)	ARS	252
	264 CONTINUE	ARS	253
	TEMP=DOT(WB, COL4)	ARS	254
	INEXT=SIGN(1.0, TEMP)	ARS	255
	IF(ILEAVE*INEXT.GE.0)GOTO 261	ARS	256
	LTRY=L	ARS	257
	INDEX=INDEX-NE	ARS	258
	GOTO 270	ARS	259
C		ARS	260
C	PROCEED TO FORGET FROM INDEX THRU NEXT ENTRY WITH DIFFERENT S	ARS	261
C	COMMENCING TO CHECK WITH THE L TH ENTRY	ARS	262
C		ARS	263
	273 INDEX1=LOCHTS+(LTRY-1)*NE	ARS	264
	IF(ABS(ASTER(INDEX1)-ASTER(INDEX)).GT.0.)GOTO 271	ARS	265
	LTRY=LTRY+1	ARS	266
	270 NHITS=NHITS-1	ARS	267
	IF(NHITS-1)800,210,273	ARS	268
	271 DO 272 LTRY=INDEX, LIMIT1	ARS	269
	IJK=LTRY+INDEX1-INDEX	ARS	270
	ASTER(LTRY)=ASTER(IJK)	ARS	271
	272 CONTINUE	ARS	272
	GOTO 220	ARS	273
C		ARS	274
	280 WRITE (6,902) INDEX	ARS	275
	290 SLAST=ASTER(INDEX)	ARS	276
	ILEAVE=INEXT	ARS	277
	300 CONTINUE	ARS	278
		APS	279

C	NOW CHOOSE THE HIT (THIS SECTION ALSO ENTERED FOR REENTRY)	ARS	280
C		ARS	281
400	DO 420 I=1,20	ARS	282
	I1ST=LOCPTS+(I-1)*NE	ARS	283
	I2ND=LOCPTS+I*NE	ARS	284
	IF(ASTER(I2ND).GE.PINF)GOTO 800	ARS	285
	IF(ASTER(I1ST).GE.PINF)GOTO 800	ARS	286
	IF(ABS(ASTER(I1ST)-ASTER(I2ND)).LE.1.0E-7)GOTO 420	ARS	287
	IF(DIST.LT.ASTER(I1ST))GOTO 410	ARS	288
	IF(DIST.GT.ASTER(I2ND))GOTO 420	ARS	289
410	K=(MASTER(IBIAS)+1)/2	ARS	290
	IF(2*K-I-MASTER(IBIAS))500,510,510	ARS	291
420	CONTINUE	ARS	292
C		ARS	293
500	RIN=ASTER(I1ST)	ARS	294
	ROUT=ASTER(I2ND)	ARS	295
	GOTO 810	ARS	296
510	RIN=ASTER(I2ND)	ARS	297
	IJK=I2ND+NE	ARS	298
	ROUT=ASTER(IJK)	ARS	299
	GOTO 810	ARS	300
800	RIN=-PINF	ARS	301
	ROUT=0.	ARS	302
810	IF(NASC.GT.-2)RETURN	ARS	303
	DO 820 I=1,84	ARS	304
	IJK=LOCPTS+I-1	ARS	305
	ASTER(IJK)=SAVE(I)	ARS	306
820	CONTINUE	ARS	307
	RETURN	ARS	308
	END	ARS	309
C		ARS	310
C		ARS	311
	SUBROUTINE BOX	****	31
	DIMENSION MASTER(30000)	BOX	2
	COMMON ASTER(30000)	BOX	3
	COMMON/PAREM/XB(3),WB(3),IR	BOX	4
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	BOX	5
	COMMON/UNCDEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	BOX	6
1	LDATA,LKIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	BOX	7
	EQUIVALENCE (MASTER,ASTER)	BOX	8
C		BOX	9
	CALL UN2(LOCDA,IV,IH1)	BOX	10
	LOC=LOCDA+1	BOX	11
	CALL UN2(LOC,IH2,IH3)	BOX	12
	RIN=-PINF	BOX	13
	ROUT=PINF	BOX	14
	DO 105 I=1,3	BOX	15
	IF(I-2)11,12,13	BOX	16
11	II=2	BOX	17
	GOTO 14	BOX	18
12	II=1	BOX	19
	GOTO 14	BOX	20
13	II=3	BOX	21
14	A=0.	BOX	22
	VP=0.	BOX	23
	W=0.	BOX	24
	DO 15 J=1,3	BOX	25
	JV=IV+J	BOX	26
	JA=IH1+J	BOX	27
	VP=VP+(ASTER(JV-1)-XB(J))*ASTER(JA-1)	BOX	28

```

      W=W+WB(J)*ASTER(JA-1)
      A=A+ASTER(JA-1)**2
15  CONTINUE
      IF(W)30,20,40
20  IF(-VP.LT.0.)GOTO 200
      IF(-VP-A)100,100,200
30  CP=VP/W
      LO=2*II-1
      IF(CP.LE.0.)GOTO 200
      CM=(VP+A)/W
      LI=LO+1
      GOTO 60
40  CP=(VP+A)/W
      LO=2*II
      IF(CP.LE.0.)GOTO 200
      CM=VP/W
      LI=LO-1
60  IF(ROUT.LE.CP)GOTO 80
      ROUT=CP
      LRO=LO
80  IF(RIN.GE.CM)GOTO 100
      RIN=CM
      LRI=LI
100 IH1=IH2
      IH2=IH3
105 CONTINUE
      IF(ABS(RIN-ROUT).LE.ROUT*1.0E-6)GOTO 200
      IF(RIN.LT.ROUT)RETURN
200 RIN=PINF
      ROUT=-PINF
      RETURN
      END

```

C
C

```

SUBROUTINE ELL
  DIMENSION FOCIA(3),FOCIB(3),MASTER(30000)
  COMMON ASTER(30000)
  COMMON/PAREM/XB(3),WB(3),IR
  COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST
  COMMON/UNCSEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,
1  LDATA,LRI,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP
  EQUIVALENCE (ASTER,MASTER)

```

C

```

CALL UN2(LOCDA,IV1,IV2)
IRR=MASTER(LOCDA+1)
FOCIA(1)=ASTER(IV1)
FOCIA(2)=ASTER(IV1+1)
FOCIA(3)=ASTER(IV1+2)
FOCIB(1)=ASTER(IV2)
FOCIB(2)=ASTER(IV2+1)
FOCIB(3)=ASTER(IV2+2)
C=ASTER(IRR)
RIN=PINF
ROUT=-PINF
D1X=XB(1)-FOCIA(1)
D1Y=XB(2)-FOCIA(2)
D1Z=XB(3)-FOCIA(3)
D2X=XB(1)-FOCIB(1)
D2Y=XB(2)-FOCIB(2)
D2Z=XB(3)-FOCIB(3)

```

BOX	29
BOX	30
BOX	31
BOX	32
BOX	33
BOX	34
BOX	35
BOX	36
BOX	37
BOX	38
BOX	39
BOX	40
BOX	41
BOX	42
BOX	43
BOX	44
BOX	45
BOX	46
BOX	47
BOX	48
BOX	49
BOX	50
BOX	51
BOX	52
BOX	53
BOX	54
BOX	55
BOX	56
BOX	57
BOX	58
BOX	59
BOX	60
BOX	61
BOX	62
****	32
ELL	2
ELL	3
ELL	4
ELL	5
ELL	6
ELL	7
ELL	8
ELL	9
ELL	10
ELL	11
ELL	12
ELL	13
ELL	14
ELL	15
ELL	16
ELL	17
ELL	18
ELL	19
ELL	20
ELL	21
ELL	22
ELL	23
ELL	24
ELL	25
ELL	26

A1=2.*(D1X*WB(1)+D1Y*WB(2)+D1Z*WB(3))	ELL	27
A2=2.*(D2X*WB(1)+D2Y*WB(2)+D2Z*WB(3))	ELL	28
B1=D1X*D1X+D1Y*D1Y+D1Z*D1Z	ELL	29
B2=D2X*D2X+D2Y*D2Y+D2Z*D2Z	ELL	30
AA=(A2-A1)/(2.*C)	ELL	31
BB=(C*C+B2-B1)/(2.*C)	ELL	32
ALAMD=AA*AA-1.	ELL	33
ALAM1=(AA*BB-.5*A2)/ALAMD	ELL	34
U=(BB*BB-B2)/ALAMD	ELL	35
DISCRM=ALAM1*ALAM1-U	ELL	36
IF(DISCRI.E.O.)RETURN	ELL	37
SQRTDI=SQRT(DISCRI)	ELL	38
RIN=-ALAM1-SQRTDI	ELL	39
ROUT=-ALAM1+SQRTDI	ELL	40
RETURN	ELL	41
END	ELL	42
	ELL	43
	ELL	44
SUBROUTINE RAW	***	33
DIMENSION H1(3),H2(3),H3(3),V(3),ASQ(3),PV(4),G(3)	RAW	2
COMMON ASTER(30000)	RAW	3
COMMON/PAREM/XB(3),WB(3),IR	RAW	4
COMMON/GEOM/LBASE,KIN,ROUT,LRI,LRO,PINF,IERR,DIST	RAW	5
COMMON/UNCSEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	RAW	6
1 LDATA,LRI,LR0T,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	RAW	7
	RAW	8
CALL UN2(LOCDA,IV,IH1)	RAW	9
LOC=LOCDA+1	RAW	10
CALL UN2(LOC,IH2,IH3)	RAW	11
H1(1)=ASTER(IH1)	RAW	12
H1(2)=ASTER(IH1+1)	RAW	13
H1(3)=ASTER(IH1+2)	RAW	14
H2(1)=ASTER(IH2)	RAW	15
H2(2)=ASTER(IH2+1)	RAW	16
H2(3)=ASTER(IH2+2)	RAW	17
H3(1)=ASTER(IH3)	RAW	18
H3(2)=ASTER(IH3+1)	RAW	19
H3(3)=ASTER(IH3+2)	RAW	20
V(1)=ASTER(IV)	RAW	21
V(2)=ASTER(IV+1)	RAW	22
V(3)=ASTER(IV+2)	RAW	23
RIN=-PINF	RAW	24
ROUT=PINF	RAW	25
CM=-PINF	RAW	26
CP=PINF	RAW	27
L=0	RAW	28
L1=0	RAW	29
K=0	RAW	30
LRI=0	RAW	31
LRO=0	RAW	32
ASQ(1)=H1(1)*H1(1)+H1(2)*H1(2)+H1(3)*H1(3)	RAW	33
ASQ(2)=H2(1)*H2(1)+H2(2)*H2(2)+H2(3)*H2(3)	RAW	34
ASQ(3)=H3(1)*H3(1)+H3(2)*H3(2)+H3(3)*H3(3)	RAW	35
XB1V1=XB(1)-V(1)	RAW	36
XB2V2=XB(2)-V(2)	RAW	37
XB3V3=XB(3)-V(3)	RAW	38
PV(1)=XB1V1*H1(1)+XB2V2*H1(2)+XB3V3*H1(3)	RAW	39
PV(2)=XB1V1*H2(1)+XB2V2*H2(2)+XB3V3*H2(3)	RAW	40
PV(3)=XB1V1*H3(1)+XB2V2*H3(2)+XB3V3*H3(3)	RAW	41
G(1)=WB(1)*H1(1)+WB(2)*H1(2)+WB(3)*H1(3)	RAW	42

G(2)=WB(1)*H2(1)+WB(2)*H2(2)+WB(3)*H2(3)	RAW	43
G(3)=WB(1)*H3(1)+WB(2)*H3(2)+WB(3)*H3(3)	RAW	44
C	RAW	45
DO 140 I=1,2	RAW	46
IF(G(I))10,110,60	RAW	47
C	RAW	48
10 IF(-PV(I))20,400,400	RAW	49
20 TEMP=-PV(I)/G(I)	RAW	50
IF(TEMP-CP)30,130,130	RAW	51
30 CP=TEMP	RAW	52
L=I	RAW	53
GOTO(40,50),I	RAW	54
40 LR0=3	RAW	55
GOTO 130	RAW	56
50 LR0=1	RAW	57
GOTO 130	RAW	58
C	RAW	59
60 IF(-PV(I).LE.0.)GOTO 130	RAW	60
TEMP=-PV(I)/G(I)	RAW	61
IF(TEMP.LE.CM)GOTO 130	RAW	62
CM=TEMP	RAW	63
K=I	RAW	64
GOTO(90,100),I	RAW	65
90 LRI=3	RAW	66
GOTO 130	RAW	67
100 LRI=1	RAW	68
GOTO 130	RAW	69
C	RAW	70
110 IF(PV(I).LE.0.)GOTO 810	RAW	71
IF(PV(I).GE.ASQ(I))GOTO 810	RAW	72
130 LI=LI+I	RAW	73
140 CONTINUE	RAW	74
C	RAW	75
IF(G(3))150,210,230	RAW	76
150 TEMP=ASQ(3)-PV(3)	RAW	77
IF(TEMP.GE.0.)GOTO 180	RAW	78
TEMP=TEMP/G(3)	RAW	79
IF(TEMP.LE.CM)GOTO 190	RAW	80
CM=TEMP	RAW	81
K=3	RAW	82
LRI=6	RAW	83
180 IF(-PV(3))190,400,400	RAW	84
190 TEMP=-PV(3)/G(3)	RAW	85
IF(TEMP.GE.CP)GOTO 290	RAW	86
CP=TEMP	RAW	87
L=3	RAW	88
LR0=5	RAW	89
GOTO 290	RAW	90
C	RAW	91
210 IF(PV(3).LE.0.)GOTO 400	RAW	92
IF(PV(3)-ASQ(3))290,290,400	RAW	93
C	RAW	94
230 IF(-PV(3).LE.0.)GOTO 260	RAW	95
TEMP=-PV(3)/G(3)	RAW	96
IF(TEMP.LE.CM)GOTO 260	RAW	97
CM=TEMP	RAW	98
K=3	RAW	99
LRI=5	RAW	100
260 TEMP=ASQ(3)-PV(3)	RAW	101
IF(TEMP.LE.0.)GOTO 400	RAW	102

TEMP=TEMP/G(3)	RAW 103
IF(TEMP.GE.CP)GOTO 290	RAW 104
CP=TEMP	RAW 105
L=3	RAW 106
LRO=6	RAW 107
290 AG=ASQ(2)*G(1)+ASQ(1)*G(2)	RAW 108
PV(4)=PV(1)*ASQ(2)+PV(2)*ASQ(1)	RAW 109
TOP=ASQ(1)*ASQ(2)-PV(4)	RAW 110
IF(AG)310,350,330	RAW 111
310 TEMP=TOP/AG	RAW 112
IF(TEMP.LE.CM)GOTO 380	RAW 113
CM=TEMP	RAW 114
K=4	RAW 115
LRI=2	RAW 116
GOTO 380	RAW 117
C	RAW 118
330 IF(TOP.LT.0.)GOTO 400	RAW 119
TEMP=TOP/AG	RAW 120
IF(TEMP-CP)370,380,380	RAW 121
C	RAW 122
350 IF(PV(4).LE.0.)GOTO 400	RAW 123
IF(-TOP)380,400,400	RAW 124
370 CP=TEMP	RAW 125
L=4	RAW 126
LKO=2	RAW 127
380 IF(L+K.LE.0)GOTO 400	RAW 128
ROUT=CP	RAW 129
RIN=CM	RAW 130
C	RAW 131
400 IF(ROUT.GE.PINF)GOTO 810	RAW 132
IF(ROUT.LE.0.)GOTO 810	RAW 133
IF(RIN.GE.ROUT)GOTO 810	RAW 134
IF(ABS(RIN-ROUT).GT.ROUT*1.0E-5)GOTO 820	RAW 135
C	RAW 136
810 ROUT=-PINF	RAW 137
RIN=PINF	RAW 138
LRO=0	RAW 139
LRI=0	RAW 140
820 RETURN	RAW 141
END	RAW 142
C	RAW 143
C	RAW 144
SUBROUTINE RCC	**** 34
DIMENSION V(3),H(3),MASTER(30000)	RCC 2
COMMON ASTER(30000)	RCC 3
COMMON/PAREM/XB(3),WB(3),IR	RCC 4
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	RCC 5
COMMON/UNCDEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	RCC 6
1 LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	RCC 7
EQUIVALENCE (ASTER,MASTER)	RCC 8
C	RCC 9
CALL UN2(LOCDA,IV,IH)	RCC 10
IRR=MASTER(LOCDA+1)	RCC 11
H(1)=ASTER(IH)	RCC 12
H(2)=ASTER(IH+1)	RCC 13
H(3)=ASTER(IH+2)	RCC 14
V(1)=ASTER(IV)	RCC 15
V(2)=ASTER(IV+1)	RCC 16
V(3)=ASTER(IV+2)	RCC 17
R=ASTER(IRR)	RCC 18

RIN=-PINF	RCC	19
ROUT=PINF	RCC	20
RSQ=R*R	RCC	21
LRO=0	RCC	22
LRI=0	RCC	23
TOP=0.	RCC	24
POT=0.	RCC	25
HH=H(1)*H(1)+H(2)*H(2)+H(3)*H(3)	RCC	26
VPH=H(1)*(V(1)-XB(1))+H(2)*(V(2)-XB(2))+H(3)*(V(3)-XB(3))	RCC	27
WH=WB(1)*H(1)+WB(2)*H(2)+WB(3)*H(3)	RCC	28
DEN=HH-WH*WH	RCC	29
DO 10 I=1,3	RCC	30
TOP=TOP+WB(I)*(XB(I)-V(I))	RCC	31
POT=POT+(XB(I)-V(I))**2	RCC	32
10 CONTINUE	RCC	33
AMBD=-HH*TOP-WH*VPH	RCC	34
UM=(POT-RSQ)*HH-VPH**2	RCC	35
IF(WH)40,70,50	RCC	36
40 CP=VPH/WH	RCC	37
CM=(VPH+HH)/WH	RCC	38
LCP=1	RCC	39
LCM=2	RCC	40
GOTO 60	RCC	41
50 CP=(VPH+HH)/WH	RCC	42
CM=VPH/WH	RCC	43
LCM=1	RCC	44
LCP=2	RCC	45
60 IF(CP)300,80,80	RCC	46
70 CP=PINF	RCC	47
CM=-CP	RCC	48
IF(VPH.GT.0.)GOTO 300	RCC	49
IF(HH+VPH)300,90,90	RCC	50
80 IF(ABS(DEN).GE.1.0E-6)GOTO 90	RCC	51
R1=-PINF	RCC	52
R2=PINF	RCC	53
GOTO 100	RCC	54
90 R1=0.	RCC	55
R2=0.	RCC	56
AMBDA=AMBD/DEN	RCC	57
UMU=UM/DEN	RCC	58
DISC=AMBDA**2-UMU	RCC	59
IF(DISC.LE.0.)GOTO 300	RCC	60
SD=SQRT(DISC)	RCC	61
R1=AMBDA-SD	RCC	62
R2=AMBDA+SD	RCC	63
100 IF(CM.GT.R1)GOTO 110	RCC	64
RIN=R1	RCC	65
LRI=3	RCC	66
GOTO 120	RCC	67
110 RIN=CM	RCC	68
LRI=LCM	RCC	69
120 IF(CP.LE.R2)GOTO 130	RCC	70
ROUT=R2	RCC	71
LRO=3	RCC	72
GOTO 200	RCC	73
130 ROUT=CP	RCC	74
LRO=LCP	RCC	75
200 IF(ABS(ROUT-RIN).LE.ROUT*1.0E-5)GOTO 300	RCC	76
GOTO(210,210,220),LRO	RCC	77
210 F1=DEN*ROUT**2-2.*AMBD*ROUT+UM	RCC	78

IF(F1)250,250,300	RCC	79
220 F1=ROUT*WH-VPH	RCC	80
IF(F1)300,250,230	RCC	81
GOTO 230	RCC	82
230 IF(F1.GI.HH) GOTO 300	RCC	83
250 GOTO(260,260,270),LRI	RCC	84
260 F1=DEN*RIN**2-2.*AMBD*RIN+UM	RCC	85
IF(F1)310,310,300	RCC	86
270 F1=RIN*WH-VPH	RCC	87
IF(F1)300,310,280	RCC	88
GOTO 280	RCC	89
280 IF(F1.LF.HH)GOTO 310	RCC	90
300 RIN=PINF	RCC	91
ROUT=-PINF	RCC	92
LRO=0	RCC	93
LRI=0	RCC	94
310 RETURN	RCC	95
END	RCC	96
C	RCC	97
C	RCC	98
SUBROUTINE REC	****	35
DIMENSION V(3),H(3),A(3),B(3)	REC	2
COMMON ASTER(30000)	REC	3
COMMON/PAREM/XB(3),WB(3),IR	REC	4
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	REC	5
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LIRIP,LSCAL,LREGD,	REC	6
1 LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	REC	7
C	REC	8
CALL UN2(LOCDA,IV,IH)	REC	9
LOC=LOCDA+1	REC	10
CALL UN2(LOC,IA,IB)	REC	11
V(1)=ASTER(IV)	REC	12
V(2)=ASTER(IV+1)	REC	13
V(3)=ASTER(IV+2)	REC	14
H(1)=ASTER(IH)	REC	15
H(2)=ASTER(IH+1)	REC	16
H(3)=ASTER(IH+2)	REC	17
A(1)=ASTER(IA)	REC	18
A(2)=ASTER(IA+1)	REC	19
A(3)=ASTER(IA+2)	REC	20
B(1)=ASTER(IB)	REC	21
B(2)=ASTER(IB+1)	REC	22
B(3)=ASTER(IB+2)	REC	23
RIN=-PINF	REC	24
ROUT=PINF	REC	25
LRO=0	REC	26
LRI=0	REC	27
AA=A(1)*A(1)+A(2)*A(2)+A(3)*A(3)	REC	28
BB=B(1)*B(1)+B(2)*B(2)+B(3)*B(3)	REC	29
V1XB1=V(1)-XB(1)	REC	30
V2XB2=V(2)-XB(2)	REC	31
V3XB3=V(3)-XB(3)	REC	32
VPA=V1XB1*A(1)+V2XB2*A(2)+V3XB3*A(3)	REC	33
VPB=V1XB1*B(1)+V2XB2*B(2)+V3XB3*B(3)	REC	34
WBA=WB(1)*A(1)+WB(2)*A(2)+WB(3)*A(3)	REC	35
WBB=WB(1)*B(1)+WB(2)*B(2)+WB(3)*B(3)	REC	36
WBAWBA=WBA*WBA	REC	37
WBBWBB=WBB*WBB	REC	38
AAAA=AA*AA	REC	39
BBBB=BB*BB	REC	40

AMBD=WBA*VPA*BBBB+WB*VPB*AAAA	REC 41
UM=BBBB*VPA*VPA+AAAA*VPR*VPB-AAAA*BBBB	REC 42
DEN=WBWBA*BUUB+WBWBB*AAAA	REC 43
IF(ABS(DEN).LE.1.0E-6)GOTO 10	REC 44
AMBUA=AMBD/DEN	REC 45
UMU=UM/DEN	REC 46
DISC=AMBUA**2-UMU	REC 47
IF(DISC.LE.0.)GOTO 300	REC 48
SD=SQRT(DISC)	REC 49
R1=AMBUA-SD	REC 50
R2=AMBUA+SD	REC 51
GOTO 20	REC 52
10 R1=-PINF	REC 53
R2=PINF	REC 54
20 HH=H(1)*H(1)+H(2)*H(2)+H(3)*H(3)	REC 55
WH=WB(1)*H(1)+WB(2)*H(2)+WB(3)*H(3)	REC 56
VPH=V1XB1*H(1)+V2XB2*H(2)+V3XB3*H(3)	REC 57
IF(WH)40,70,50	REC 58
40 IF(VPH.GE.0.)GOTO 300	REC 59
CP=VPH/WH	REC 60
CM=(VPH+HH)/WH	REC 61
LCP=1	REC 62
LCM=2	REC 63
GOTO 100	REC 64
50 VPHHH=VPH+HH	REC 65
IF(VPHHH.LE.0.)GOTO 300	REC 66
CP=VPHHH/WH	REC 67
CM=VPH/WH	REC 68
LCM=1	REC 69
LCP=2	REC 70
GOTO 100	REC 71
70 CP=PINF	REC 72
CM=-CP	REC 73
100 IF(CM.GT.R1)GOTO 110	REC 74
RIN=R1	REC 75
LRI=3	REC 76
GOTO 120	REC 77
110 RIN=CM	REC 78
LRI=LCM	REC 79
120 IF(CP.LE.R2)GOTO 130	REC 80
ROUT=R2	REC 81
LRO=3	REC 82
GOTO 200	REC 83
130 ROUT=CP	REC 84
LRO=LCP	REC 85
200 IF(ABS(ROUT-RIN).LE.ROUT*1.0E-5)GOTO 300	REC 86
GOTO(210,210,220),LRO	REC 87
210 F1=DEN*ROUT**2-2.*AMBD*ROUT+UM	REC 88
IF(F1)250,250,300	REC 89
220 F1=ROUT*WH-VPH	REC 90
IF(F1)300,250,230	REC 91
GOTO 230	REC 92
230 IF(F1.GT.HH)GOTO 300	REC 93
250 GOTO(260,260,270),LRI	REC 94
260 F1=DEN*RIN**2-2.*AMBD*RIN+UM	REC 95
IF(F1)310,310,300	REC 96
270 F1=RIN*WH-VPH	REC 97
IF(F1)300,310,280	REC 98
GOTO 280	REC 99
280 IF(F1.LE.HH)GOTO 310	REC 100

300	RIN=PINF	REC	104
	ROUT=-PINF	REC	105
	LRI=0	REC	106
	LRO=0	REC	107
310	RETURN	REC	108
	END	REC	109
C		REC	110
C		REC	111
	SUBROUTINE RPP(NB0)	REC	112
	DIMENSION MASTER(30000),PR(6),LR(6),XS(6),LST(6)	REC	113
	COMMON ASTER(30000)	REC	114
	COMMON/PAREM/XB(3),WB(3),IR	REC	115
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	REC	116
	COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGO,	REC	117
	1 LUATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	REC	118
	EQUIVALENCE (MASTER,ASTER)	REC	119
C		REC	120
901	FORMAT(1H0,12HERROR IN RPP/4H L =,I10,5X,4HNBO=,I10,5X,3HIR=,	REC	121
	1 110/4H XB=,3E20.10/4H WB=,3E20.10/4H PR=,6E20.10/4H LR=,6I10)	REC	122
C		REC	123
	LST(1)=1	REC	124
	LST(2)=1	REC	125
	LST(3)=2	REC	126
	LST(4)=2	REC	127
	LST(5)=3	REC	128
	LST(6)=3	REC	129
	L=0	REC	130
	PR(1)=0.	REC	131
	PR(2)=0.	REC	132
	DO 10 I=1,6	REC	133
	XS(I)=S(NB0,I)	REC	134
10	CONTINUE	REC	135
C		REC	136
	DO 100 I=1,6	REC	137
	II=LST(I)	REC	138
	TEMP=XS(I)-XB(II)	REC	139
	IF(WB(II)) 20,100,30	REC	140
20	IF(TEMP)40,100,100	REC	141
30	IF(TEMP.LE.0.)GOTO 100	REC	142
40	TRY=TEMP/WB(II)	REC	143
	DO 60 J=1,3	REC	144
	IF(J.EQ.II)GOTO 60	REC	145
	XRY=XB(J)+TRY*WB(J)	REC	146
	IF((XS(2*J-1)-XRY)*(XRY-XS(2*J)).LT.0.)GOTO 100	REC	147
60	CONTINUE	REC	148
	L=L+1	REC	149
	PR(L)=TRY	REC	150
	LR(L)=I	REC	151
	IF(L.EQ.2)GOTO 130	REC	152
	IF(L.LT.2)GOTO 100	REC	153
	WRITE (6,901)L,NB0,IR,XB,WB,PR,LR	REC	154
	ROUT=-PINF	REC	155
	RETURN	REC	156
100	CONTINUE	REC	157
	GOTO 160	REC	158
C		REC	159
130	IF(ABS(PR(1)-PR(2)).LE.PR(1)*1.0E-6)GOTO 200	REC	160
	IF(PR(1)-PR(2))140,180,150	REC	161
140	RIN=PR(1)	REC	162
	LR=LR(1)	REC	163

ROUT=PR(2)	RPP	53
LRO=LR(2)	RPP	54
RETURN	RPP	55
150 RIN=PR(2)	RPP	56
LR1=LR(2)	RPP	57
ROUT=PR(1)	RPP	58
LRO=LR(1)	RPP	59
RETURN	RPP	60
C	RPP	61
160 IF(L.GE.1)GOTO 180	RPP	62
170 ROUT=-PINF	RPP	63
RETURN	RPP	64
180 RIN=-PINF	RPP	65
LRI=0	RPP	66
ROUT=PR(1)	RPP	67
LRO=LR(1)	RPP	68
RETURN	RPP	69
C	RPP	70
200 DO 220 J=1,3	RPP	71
IF(XB(J).LT.XS(2*J-1))GOTO 170	RPP	72
IF(XB(J).GT.XS(2*J))GOTO 170	RPP	73
220 CONTINUE	RPP	74
GOTO 180	RPP	75
END	RPP	76
C	RPP	77
C	RPP	78
SUBROUTINE RPP2(LSURF,XP,IRP)	****	37
C	RPP2	2
FINDS ABUTING RPP	RPP2	3
DIMENSION XP(3)	RPP2	4
COMMON/ASTER(30000)	RPP2	5
COMMON/PAREM/XB(3),WB(3),IR	RPP2	6
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	RPP2	7
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	RPP2	8
1 LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	RPP2	9
C	RPP2	10
LOC=LBASE+12*(NASC-1)-2*(LSURF+1)	RPP2	11
CALL UN2(LOC,LOCAT,NC)	RPP2	12
IF(NC-1)10,20,30	RPP2	13
10 IRP=0	RPP2	14
RETURN	RPP2	15
20 CALL UN2(LOCAT,IRP,DUM)	RPP2	16
RETURN	RPP2	17
30 M=1	RPP2	18
C	RPP2	19
DO 90 I=1,NC	RPP2	20
M=-M	RPP2	21
IF(M.GT.0)GOTO 50	RPP2	22
CALL UN2(LOCAT,I1,I2)	RPP2	23
LOCAT=LOCAT+1	RPP2	24
IRP=I1	RPP2	25
GOTO 70	RPP2	26
50 IRP=I2	RPP2	27
70 LS=(1-LSURF)/2	RPP2	28
DO 80 J=1,3	RPP2	29
IF(J.EQ.LS)GOTO 80	RPP2	30
IF((S(IRP,2*J-1)-XP(J))*(XP(J)-S(IRP,2*J)).LT.0.)GOTO 90	RPP2	31
80 CONTINUE	RPP2	32
RETURN	RPP2	33
90 CONTINUE	RPP2	34
IRP=0		

RETURN	RPP2	35
END	RPP2	36
C	RPP2	37
C	RPP2	38
SUBROUTINE SPH	****	38
COMMON ASTER(30000)	SPH	2
COMMON/PAREM/XB(3),WB(3),IR	SPH	3
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	SPH	4
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	SPH	5
1 LDATA,LRIN,LROI,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	SPH	6
C	SPH	7
CALL UN2(LOCDA,ITEMP,I2)	SPH	8
R=ASTER(I2)	SPH	9
ITEMP=ITEMP+1	SPH	10
DX=XB(1)-ASTER(ITEMP-1)	SPH	11
DY=XB(2)-ASTER(ITEMP)	SPH	12
DZ=XB(3)-ASTER(ITEMP+1)	SPH	13
B=DX*WB(1)+DY*WB(2)+DZ*WB(3)	SPH	14
C=DX*DX+DY*DY+DZ*DZ-R*R	SPH	15
DIS=B*B-C	SPH	16
IF(C.GT.0.)GOTO 10	SPH	17
RIN=-PINF	SPH	18
ROUT=SQRT(DIS)-B	SPH	19
RETURN	SPH	20
10 IF(DIS.GT.0.)GOTO 20	SPH	21
RIN=PINF	SPH	22
ROUT=-PINF	SPH	23
RETURN	SPH	24
20 DIS=SQRT(DIS)	SPH	25
RIN=-B-DIS	SPH	26
ROUT=-B+DIS	SPH	27
RETURN	SPH	28
END	SPH	29
C	SPH	30
C	SPH	31
SUBROUTINE TEC	****	39
DIMENSION MASTER(30000),DELTA(3),HF(3),AUN(3)	TEC	2
COMMON ASTER(30000)	TEC	3
COMMON/PAREM/XB(3),WB(3),IR	TEC	4
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	TEC	5
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	TEC	6
1 LDATA,LRIN,LROI,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	TEC	7
EQUIVALENCE (MASTER,ASTER),(GAMMA,SIGMA)	TEC	8
REAL NF(3),K(3),M,M2,MM,MM2	TEC	9
C	TEC	10
CALL UN2(LOCDA,IV,IH)	TEC	11
LOC=LOCDA+1	TEC	12
CALL UN2(LOC,IN,IA)	TEC	13
LOC=LOC+1	TEC	14
CALL UN2(LOC,IR1,IR2)	TEC	15
IRC=MASTER(LOC+1)	TEC	16
R1=ASTER(IR1)	TEC	17
R2=ASTER(IR2)	TEC	18
R3=R1/ASTER(IRC)	TEC	19
R4=R2/ASTER(IRC)	TEC	20
DDN=0.	TEC	21
WDA=0.	TEC	22
DDA=0.	TEC	23
HDA=0.	TEC	24
HDN=0.	TEC	25

WDN=0.	TEC 26
DO 100 I=1,3	TEC 27
I1=I-1	TEC 28
J1=I1+1	TEC 29
J2=I1+1	TEC 30
J3=I1+1	TEC 31
J4=I1+1	TEC 32
DELTA(I)=ASTER(J1)-XB(I)	TEC 33
HF(I)=ASTER(J2)	TEC 34
NF(I)=ASTER(J3)	TEC 35
AUN(I)=ASTER(J4)	TEC 36
DDN=DELTA(I)*NF(I)+DDN	TEC 37
WDA=WB(I)*AUN(I)+WDA	TEC 38
DDA=DELTA(I)*AUN(I)+DDA	TEC 39
HDA=HF(I)*AUN(I)+HDA	TEC 40
HON=HF(I)*NF(I)+HON	TEC 41
WDN=WB(I)*NF(I)+WDN	TEC 42
100 CONTINUE	TEC 43
CALL CROSS(K,AUN,NF)	TEC 44
WDK=DOT(WB,K)	TEC 45
DDK=DOT(DELTA,K)	TEC 46
HDK=DOT(HF,K)	TEC 47
IF(ABS(WDN).GT.1.0E-7)GOTO 300	TEC 48
GAMMA=-DDN/HON	TEC 49
IF(GAMMA.LT.0.)GOTO 900	TEC 50
RTP=GAMMA-1.	TEC 51
IF(RTP.GT.0.)GOTO 900	TEC 52
M=GAMMA*R3+R1*(1.-GAMMA)	TEC 53
MM=GAMMA*R4+R2*(1.-GAMMA)	TEC 54
M2=M*M	TEC 55
MM2=MM*MM	TEC 56
T=SIGMA*HDA+DDA	TEC 57
TT=SIGMA*HDK+DDK	TEC 58
A=MM2*WDA**2+M2*WDK**2	TEC 59
B=- (MM2*WDA*T+M2*WDK*TT)	TEC 60
C=MM2*T**2+M2*TT**2-M2*M2	TEC 61
DISC=B*B-A*C	TEC 62
IF(DISC.LT.0.)GOTO 900	TEC 63
IF(DISC.GT.0.)DISC=SQRT(DISC)	TEC 64
RIN=(-B-DISC)/A	TEC 65
ROUT=(-B+DISC)/A	TEC 66
LRI=3	TEC 67
LRO=3	TEC 68
GOTO 950	TEC 69
C	TEC 70
300 FLIPD=1.	TEC 71
IF(WDN.LT.0.)GOTO 310	TEC 72
FLIPD=-1.	TEC 73
WDA=-WDA	TEC 74
WDN=-WDN	TEC 75
WDK=-WDK	TEC 76
310 ALPHA=HON/WDN	TEC 77
BETA=DDN/WDN	TEC 78
TAU=(R3/R4)**2	TEC 79
A=(ALPHA*WDA-HDA)**2+TAU*(ALPHA*WDK-HDK)**2-TAU*(R4-R2)**2	TEC 80
B=- (-ALPHA*BETA*WDA**2+ALPHA*WDA*DDA+BETA*WDA*HDA-DDA*HDA	TEC 81
1 +TAU*(-ALPHA*BETA*WDK**2+ALPHA*WDK*DDK+BETA*WDK*HDK-DDK*HDK	TEC 82
2 +R2*R4-R2*R2))	TEC 83
C=(DDA-BETA*WDA)**2+TAU*((DDK-BETA*WDK)**2-R2**2)	TEC 84
DISC=B*B-A*C	TEC 85

```

IF(DISC.LT.0.)GOTO 900
IF(DISC.GT.0.)DISC=SQRT(DISC)
IF(ABS(A).LE.1.0E-7)GOTO 330
IF(A)320,330,340
320 SIGMA1=(-B-DISC)/A
    SIGMA2=(-B+DISC)/A
    GOTO 350
330 SIGMA1=-C/(2.*B)
    SIGMA2=-PINF
    IF(SIGMA1)900,350,350
340 SIGMA1=(-B+DISC)/A
    SIGMA2=(-B-DISC)/A
350 SIGMAP=-R1/(R3-R1)
    IF(SIGMA2.GT.1.)GOTO 900
    IF(SIGMA1.LT.0.)GOTO 900
    IF(SIGMA1.GT.1.)GOTO 410
    IF(SIGMA2.GT.0.)GOTO 400
    RIN=ALPHA*SIGMA1+BETA
    LRI=3
    ROUT=BETA
    LRO=1
    GOTO 490
400 RIN=ALPHA*SIGMA1+BETA
    LRI=3
    ROUT=ALPHA*SIGMA2+BETA
    LRO=3
    GOTO 490
410 IF(SIGMA2.GT.0.)GOTO 440
    IF(SIGMA1.GT.SIGMAP)GOTO 900
    RIN=ALPHA+BETA
    LRI=2
    ROUT=BETA
    LRO=1
    GOTO 490
440 IF(SIGMA1.GT.SIGMAP)GOTO 460
    RIN=ALPHA+BETA
    LRI=2
    ROUT=ALPHA*SIGMA2+BETA
    LRO=3
    GOTO 490
460 RIN=ALPHA*SIGMA2+BETA
    LRI=3
    ROUT=BETA
    LRO=1
C
490 IF(FLIPD.GE.0.)GOTO 950
    RTP=RIN
    ITP=LRI
    RIN=-ROUT
    LRI=LRO
    ROUT=-RTP
    LRO=ITP
    GOTO 950
900 RIN=PINF
    ROUT=-PINF
950 IF(ROUT.GT.0.)GOTO 1000
    RIN=PINF
    ROUT=-PINF
    RETURN
1000 IF(ABS(ROUT-RIN).LE.RIN*1.0E-6)GOTO 900

```

```

TEC 86
TEC 87
TEC 88
TEC 89
TEC 90
TEC 91
TEC 92
TEC 93
TEC 94
TEC 95
TEC 96
TEC 97
TEC 98
TEC 99
TEC 100
TEC 101
TEC 102
TEC 103
TEC 104
TEC 105
TEC 106
TEC 107
TEC 108
TEC 109
TEC 110
TEC 111
TEC 112
TEC 113
TEC 114
TEC 115
TEC 116
TEC 117
TEC 118
TEC 119
TEC 120
TEC 121
TEC 122
TEC 123
TEC 124
TEC 125
TEC 126
TEC 127
TEC 128
TEC 129
TEC 130
TEC 131
TEC 132
TEC 133
TEC 134
TEC 135
TEC 136
TEC 137
TEC 138
TEC 139
TEC 140
TEC 141
TEC 142
TEC 143
TEC 144
TEC 145

```

RETURN	TEC 146
END	TEC 147
C	TEC 148
C	TEC 149
SUBROUTINE TOR	**** 40
DIMENSION MASTER(30000),XMCV(3),C(4),RT(4),RTS(4),XAW(3),XTRY(3)	TOR 2
COMMON ASTER(30000)	TOR 3
COMMON/PAREM/XB(3),WB(3),IR	TOR 4
COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	TOR 5
COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	TOR 6
! LDATA,LRI,LR0T,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	TOR 7
EQUIVALENCE (MASTER,ASTER),(DIST,STHUS)	TOR 8
REAL NF(3)	TOR 9
C	TOR 10
CALL UN2(LOCDA,IV,IN)	TOR 11
LOC=LOCDA+1	TOR 12
CALL UN2(LOC,IR1,IR2)	TOR 13
R1=ASTER(IR1)	TOR 14
R2=ASTER(IR2)	TOR 15
WDN=0.	TOR 16
XMC2=0.	TOR 17
AW=SQRT(DOT(WB,WB))	TOR 18
DO 10 I=1,3	TOR 19
J1=IV+I-1	TOR 20
XAW(I)=ASTER(J1)-XB(I)	TOR 21
10 CONTINUE	TOR 22
RSAVE=ABS(DOT(XAW,WB)/AW)-R1-R2-R2	TOR 23
IF(NASC.EQ.-2)RSAVE=0.	TOR 24
DO 20 I=1,3	TOR 25
XTRY(I)=XB(I)+RSAVE*WB(I)	TOR 26
20 CONTINUE	TOR 27
DO 100 I=1,3	TOR 28
J1=IV+I-1	TOR 29
J2=IN+I-1	TOR 30
NF(I)=ASTER(J2)	TOR 31
XMCV(I)=XTRY(I)-ASTER(J1)	TOR 32
XMC2=XMCV(I)**2+XMC2	TOR 33
WDN=WB(I)*ASTER(J2)+WDN	TOR 34
100 CONTINUE	TOR 35
WDXMC=DOT(WB,XMCV)	TOR 36
XMCDN=DOT(XMCV,NF)	TOR 37
R12=R1*R1	TOR 38
R22=R2*R2	TOR 39
TERM=R12+R22-XMC2	TOR 40
C(1)=4.*WDXMC	TOR 41
TEMP=4.*WDXMC**2	TOR 42
C(2)=4.*R12*WDN**2-2.*TERM+TEMP	TOR 43
C(3)=8.*R12*WDN*XMCDN-4.*WDXMC*TERM	TOR 44
C(4)=4.*R12*(XMCDN**2-R22)+TERM**2	TOR 45
CALL QRTIC(C,RT,NR)	TOR 46
IF(NR-2)110,120,140	TOR 47
C	TOR 48
TOR NOT HIT	TOR 49
110 RIN=0.	TOR 50
ROUT=-PINF	TOR 51
RETURN	TOR 52
C	TOR 53
2 ROOTS	TOR 54
120 IF(RT(1).GE.RT(2))GOTO 130	TOR 55
RIN=RT(1)	TOR 56
ROUT=RT(2)	
GOTO 900	

130	RIN=RT(2)	TOR	57
	ROUT=RT(1)	TOR	58
	GOTO 900	TOR	59
C	4 ROOTS SELECT FIRST PAIR .GE. DIST AS RIN AND ROUT	TOR	60
140	RTS(1)=RT(1)	TOR	61
	IF(RT(2).LT.RTS(1))GOTO 150	TOR	62
	RTS(2)=RT(2)	TOR	63
	GOTO 160	TOR	64
150	RTS(2)=RTS(1)	TOR	65
	RTS(1)=RT(2)	TOR	66
160	IF(RT(3).LT.RTS(2))GOTO 170	TOR	67
	RTS(3)=RT(3)	TOR	68
	GOTO 190	TOR	69
170	RTS(3)=RTS(2)	TOR	70
	IF(RT(3).LT.RTS(1))GOTO 180	TOR	71
	RTS(2)=RT(3)	TOR	72
	GOTO 190	TOR	73
180	RTS(2)=RTS(1)	TOR	74
	RTS(1)=RT(3)	TOR	75
190	IF(RT(4).LT.RTS(3))GOTO 200	TOR	76
	RTS(4)=RT(4)	TOR	77
	GOTO 300	TOR	78
200	RTS(4)=RTS(3)	TOR	79
	IF(RT(4).LT.RTS(2))GOTO 210	TOR	80
	RTS(3)=RT(4)	TOR	81
	GOTO 300	TOR	82
210	RTS(3)=RTS(2)	TOR	83
	IF(RT(4).LT.RTS(1))GOTO 220	TOR	84
	RTS(2)=RT(4)	TOR	85
	GOTO 300	TOR	86
220	RTS(2)=RTS(1)	TOR	87
	RTS(1)=RT(4)	TOR	88
C	STHUS=DIST	TOR	89
300	IF(ABS(STHUS-RTS(2)).LE.1.0E-7)GOTO 310	TOR	90
	IF(STHUS.GE.RTS(2))GOTO 310	TOR	91
	RIN=RTS(1)	TOR	92
	ROUT=RTS(2)	TOR	93
	GOTO 900	TOR	94
310	RIN=RTS(3)	TOR	95
	ROUT=RTS(4)	TOR	96
C		TOR	97
900	LRI=1	TOR	98
	LRO=1	TOR	99
	RIN=RIN+RSAVE	TOR	100
	ROUT=ROUT+RSAVE	TOR	101
	IF(ROUT.GE.0.0)GOTO 920	TOR	102
910	RIN=PINF	TOR	103
	ROUT=-PINF	TOR	104
	RETURN	TOR	105
920	IF(ABS(ROUT-RIN).LE.RIN*1.0E-6)GOTO 910	TOR	106
	RETURN	TOR	107
	END	TOR	108
C		TOR	109
C		TOR	110
	SUBROUTINE TRC	****	41
	DIMENSION MASTER(30000),V(3),H(3)	TRC	2
	COMMON ASTER(30000)	TRC	3
	COMMON/PAREM/XB(3),WB(3),IR	TRC	4
	COMMON/GEOM/LBASE,RIN,ROUT,LRI,LRO,PINF,IERR,DIST	TRC	5
	COMMON/UNCGEM/NRPP,NTRIP,NSCAL,NBODY,NRMAX,LTRIP,LSCAL,LREGD,	TRC	6

1	LDATA,LRIN,LROT,LIO,LOCDA,I15,I30,LBODY,NASC,KLOOP	TRC	7
	EQUIVALENCE(MASTER,ASTER)	TRC	8
C		TRC	9
	CALL UN2(LOCDA,IV,IH)	TRC	10
	LOC=LOCDA+1	TRC	11
	CALL UN2(LOC,IRB,IRTOP)	TRC	12
	V(1)=ASTER(IV)	TRC	13
	V(2)=ASTER(IV+1)	TRC	14
	V(3)=ASTER(IV+2)	TRC	15
	H(1)=ASTER(IH)	TRC	16
	H(2)=ASTER(IH+1)	TRC	17
	H(3)=ASTER(IH+2)	TRC	18
	RB=ASTER(IRB)	TRC	19
	RT=ASTER(IRTOP)	TRC	20
	RIN=-PINF	TRC	21
	ROUT=PINF	TRC	22
	LRO=0	TRC	23
	LRI=0	TRC	24
	INTSEC=0	TRC	25
	INTR1=0	TRC	26
	INTR2=0	TRC	27
	V1XB1=V(1)-XB(1)	TRC	28
	V2XB2=V(2)-XB(2)	TRC	29
	V3XB3=V(3)-XB(3)	TRC	30
	PVPV=V1XB1*V1XB1+V2XB2*V2XB2+V3XB3*V3XB3	TRC	31
	VPW=V1XB1*WB(1)+V2XB2*WB(2)+V3XB3*WB(3)	TRC	32
	WH=WB(1)*H(1)+WB(2)*H(2)+WB(3)*H(3)	TRC	33
	VPH=V1XB1*H(1)+V2XB2*H(2)+V3XB3*H(3)	TRC	34
	HH=H(1)*H(1)+H(2)*H(2)+H(3)*H(3)	TRC	35
	RTRB=RT-RB	TRC	36
	RBRTVP=RB-VPH*RTRB/HH	TRC	37
	VPHHH=VPH+HH	TRC	38
	UM=HH*(PVPV-RBRTVP**2)-VPH*VPH	TRC	39
	AMBD=HH*VPW-WH*(VPH-RTRB*RBRTVP)	TRC	40
	DEN=HH-WH**2*(1.+RTRB**2/HH)	TRC	41
	IF(ABS(DEN).GT.1.0E-6)GOTO 40	TRC	42
	IF(RTRB.EQ.0.)GOTO 200	TRC	43
	R2=UM/(2.*AMBD)	TRC	44
	F1=R2*WH-VPH	TRC	45
	IF(F1.LT.0.)GOTO 200	TRC	46
	IF(F1.GT.HH)GOTO 200	TRC	47
	INTSEC=INTSEC+1	TRC	48
	IF(WH.LE.0.)GOTO 10	TRC	49
	IF(RTRB)20,20,30	TRC	50
10	IF(RTRB)30,30,20	TRC	51
20	LRO=3	TRC	52
	ROUT=R2	TRC	53
	GOTO 250	TRC	54
30	LRI=3	TRC	55
	RIN=R2	TRC	56
	INTSEC=INTSEC+1	TRC	57
	GOTO 210	TRC	58
C		TRC	59
40	AMBDA=AMBD/DEN	TRC	60
	UMU=UM/DEN	TRC	61
	DISC=AMBDA**2-UMU	TRC	62
	IF(DISC)350,200,50	TRC	63
	GOTO 50	TRC	64
50	SD=SQRT(DISC)	TRC	65
	R1=AMBDA-SD	TRC	66

R2=AMBDA+SD	TRC	67
F1=R2*WH-VPH	TRC	68
IF(F1.LT.0.)GOTO 60	TRC	69
IF(F1.LE.HH)INTR2=INTR2+1	TRC	70
60 F1=R1*WH-VPH	TRC	71
IF(F1.LT.0.)GOTO 70	TRC	72
IF(F1.LE.HH)GOTO 80	TRC	73
70 IF(INTR2.LT.1)GOTO 200	TRC	74
ROUT=R2	TRC	75
RIN=R2	TRC	76
LRO=3	TRC	77
LRI=3	TRC	78
INTSEC=INTSEC+1	TRC	79
GOTO 200	TRC	80
80 INTR1=INTR1+1	TRC	81
IF(INTR2.GE.1)GOTO 90	TRC	82
ROUT=R1	TRC	83
RIN=R1	TRC	84
LRO=3	TRC	85
LRI=3	TRC	86
INTSEC=INTSEC+1	TRC	87
GOTO 200	TRC	88
90 IF(R1-R2)100,350,110	TRC	89
100 RIN=R1	TRC	90
ROUT=R2	TRC	91
LRO=3	TRC	92
LRI=3	TRC	93
GOTO 300	TRC	94
110 RIN=R2	TRC	95
ROUT=R1	TRC	96
LRO=3	TRC	97
LRI=3	TRC	98
GOTO 300	TRC	99
C	TRC	100
200 IF(WH)210,350,250	TRC	101
210 IF(VPH.GE.0.)GOTO 350	TRC	102
CP=VPH/WH	TRC	103
F1=CP*CP-2.*CP*VPW+PVPV-RB*RB	TRC	104
IF(F1.GT.0.)GOTO 220	TRC	105
INTSEC=INTSEC+1	TRC	106
ROUT=CP	TRC	107
LRO=1	TRC	108
IF(INTSEC.GE.2)GOTO 300	TRC	109
220 CM=VPHHH/WH	TRC	110
F1=CM*CM-2.*((VPW+WH)*CM-VPH)+HH+PVPV-RT*RT	TRC	111
IF(F1.GT.0.)GOTO 350	TRC	112
RIN=CM	TRC	113
LRI=2	TRC	114
GOTO 300	TRC	115
250 IF(VPHHH.LT.0.)GOTO 350	TRC	116
CP=VPHHH/WH	TRC	117
F1=CP*CP-2.*((VPW+WH)*CP-VPH)+HH+PVPV-RT*RT	TRC	118
IF(F1.GT.0.)GOTO 260	TRC	119
INTSEC=INTSEC+1	TRC	120
ROUT=CP	TRC	121
LRO=2	TRC	122
260 IF(INTSEC.GE.2)GOTO 300	TRC	123
CM=VPH/WH	TRC	124
F1=CM*CM-2.*CP*VPW+PVPV-RB*RB	TRC	125
IF(F1.GT.0.)GOTO 350	TRC	126

```

      RIN=CM
      LRI=1
C
300 IF (ABS(ROUT-RIN)-ROUT*1.0E-5)350,350,360
350 RIN=PINF
      ROUT=-PINF
      LRI=0
      LRO=0
360 RETURN
      END
L
C
END

```

```

TRC 127
TRC 128
TRC 129
TRC 130
TRC 131
TRC 132
TRC 133
TRC 134
TRC 135
TRC 136
TRC 137
TRC 138
TRC 139

```